

IVO MARIO MATHIAS

**SISMAT**  
**SISTEMA DE MATRÍCULA INTELIGENTE**

Dissertação apresentada como requisito parcial  
à obtenção do título de Mestre em Ciências,  
Curso de Pós-Graduação em Informática, Univer-  
sidade Federal do Paraná.

Orientadora:  
Dr.<sup>a</sup> Aurora Trinidad Ramirez Pozo

Curitiba - Paraná  
2000

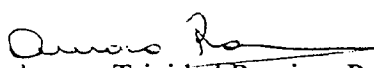


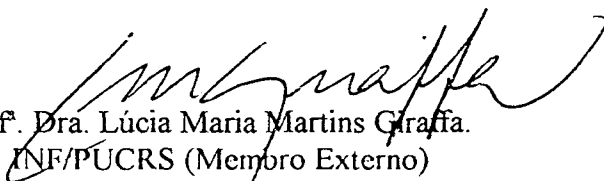
Ministério da Educação  
Universidade Federal do Paraná  
Mestrado em Informática

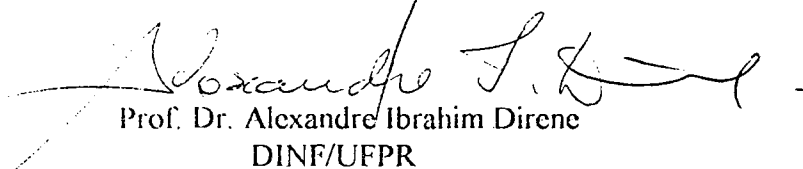
## PARECER

Nós, abaixo assinados, membros da Comissão Examinadora da defesa de Dissertação de Mestrado em Informática, do aluno Ivo Mário Mathias, avaliamos o trabalho intitulado **“SISMAT - Sistema de Matrícula Inteligente”**, Linha de Pesquisa - Inteligência Computacional, cuja defesa foi realizada no dia 17 de julho de 2000. Após a Avaliação, decidimos pela Aprovação do Candidato.

Curitiba, 17 de julho de 2000.

  
Prof.<sup>a</sup> Dra. Aurora-Trinidad Ramirez Pozo  
Presidente DINF/UFPR

  
Prof.<sup>a</sup> Dra. Lúcia Maria Martins Graffa.  
INF/PUCRS (Membro Externo)

  
Prof. Dr. Alexandre Ibrahim Direne  
DINF/UFPR

### **DEDICATÓRIA**

Dedico esta pesquisa aos meus pais Estanislau e Catarina, pelo empenho e ajuda que sempre me deram, estimulando-me e propiciando-me muitos conhecimentos e realizações, reerguendo sempre o meu ânimo e a minha vontade de aprender.

## AGRADECIMENTOS

Agradeço a Deus, que sempre esteve presente em todos os momentos de minha vida, fornecendo-me idéias que favoreceram a realização deste objetivo, aprender. E, sem a força deste, nada seria possível.

Agradeço a Marília, minha esposa, que sempre me estimulou, nos momentos de incerteza, e me compreendeu, nos momentos de dedicação à pesquisa. Obrigado por seu amor, sua tolerância e sua paciência.

Um agradecimento muito especial a minhas filhas Fernanda e Camila, que mesmo na sua inocência e pureza, souberam me entender e me estimular quando precisei. Vocês são a inspiração da minha vida.

Agradeço a minha orientadora Professora Aurora, pela incansável orientação e assistência dedicadas as atividades deste mestrado, bem como os estímulos nos momentos de incertezas.

Não poderia deixar de agradecer a minha amiga Alaine, pelo incentivo, apoio e amizade durante todo o meu mestrado.

Aos meus amigos da UEPG e UFPR pelo apoio e amizade durante esta jornada.

À UEPG pela licença remunerada que viabilizou o meu mestrado.

Meus agradecimentos à Capes, cujos recursos me auxiliaram na realização do trabalho.

Meus agradecimentos à CRIM - Centre de Recherche Informatique de Montreal, por permitirem usar a ferramenta de desenvolvimento de agentes LALO, para o desenvolvimento do experimento prático deste trabalho.

## Sumário

Lista de abreviaturas	viii
Lista de Figuras	x
Lista de Tabelas	xi
Resumo	1
Abstract	2
<b>1 Introdução</b>	<b>3</b>
1.1 Contexto do trabalho	3
1.2 Especificação do problema	4
1.3 Objetivo do Trabalho	4
1.4 Organização do Trabalho	6
<b>2 Inteligência Artificial</b>	<b>9</b>
2.1 Introdução	9
2.2 Inteligência Artificial Distribuída	10
2.2.1 Resolução Distribuída de Problemas (RDP)	12
2.2.2 Sistemas Multiagentes (SMA)	12
2.3 Conclusões	14
<b>3 Agentes</b>	<b>15</b>
3.1 Introdução	15
3.2 Definições	15
3.3 Propriedades de Agentes	17
3.4 Compartilhamento de conhecimento entre os agentes	19
3.5 Arquitetura de um agente	21
3.6 Estados Mentais	23
3.7 Ambientes de desenvolvimento de Agentes	26
3.7.1 Diferenças entre ambientes de desenvolvimento de Agentes	27
3.7.2 Exemplos de Plataformas de desenvolvimento de agentes	28
3.7.3 LALO (Langage d'Agents Logiciel Objet)	28
3.8 Conclusões	29

<b>4</b>	<b>SISMAT</b>	<b>30</b>
4.1	Introdução	30
4.2	Tarefas da sociedade de agentes do SISMAT	31
4.3	Estados Mentais - SISMAT	35
4.4	Agente Usuário (AU)	38
4.5	Agente Interface (AI)	38
4.6	Agente Mediador (AM)	40
4.7	Agente Matriculador (AMT)	44
4.8	Discussões	48
<b>5</b>	<b>Resultados: SISMAT</b>	<b>50</b>
5.1	Filosofia de Operação	50
5.2	Exemplo de Matrícula: Acadêmico sem dependência	50
5.3	Exemplo de Matrícula: Acadêmico com dependência	56
5.4	Estados mentais envolvidos em uma matrícula	60
5.4.1	Estados Mentais AI	60
5.4.2	Estados Mentais AM	63
5.4.3	Estados Mentais AMT	67
5.5	Propriedades dos agentes do SISMAT	72
5.6	Discussões	73
<b>6</b>	<b>Conclusões</b>	<b>76</b>
	<b>Referências Bibliográficas</b>	<b>78</b>

Listagem dos Programas LALO e C++ encontram-se disponíveis num módulo em separado.

Relatório Técnico – encontra-se disponível num módulo em separado, um relatório técnico que contempla aspectos operacionais e as características técnicas da ferramenta de desenvolvimento de agentes LALO (*Langage d'Agents Logiciel Objet*) versão 1.3, cujo trabalho é fruto das pesquisas realizadas durante esta dissertação.

## Lista de abreviaturas

AgentBuilder®	Marca registrada da Reticular Systems, Inc.
AOP	Agent-Oriented Programming (Programação Orientada a Agentes)
ARPA	Advanced Research Projects Agency
AI	Agente Interface
AM	Agente Mediador
AMT	Agente Matriculador
AU	Agente Usuário
BC	Base de Conhecimento
BD	Bases de Dados
HP	Hewlett Packard
IA	Inteligência Artificial
IAD	Inteligência Artificial Distribuída
IADz	Inteligência Artificial Descentralizada
IRIX	Sistema Operacional da Silicon Graphics International
JRE	Java Runtime Environment
KQML	Knowledge Query and Manipulation Language
KSE	Knowledge-Sharing Effort
LALO	Langage of Agents Logiciel Objet
LINUX	Sistema operacional compatível com o UNIX, criado por Linus Torvalds
MAS	Multi Agents Systems (Sistemas Multiagentes)
OOP	Object-Oriented Programming (Programação Orientada a Objetos)
RA	Registro Acadêmico
RDP	Resolução Distribuída de Problemas
SISMAT	Sistema de Matrícula Inteligente
SMA	Sistemas Multiagentes
SYSIR	System of Intelligent Registration (Sistema de Matrícula Inteligente)
UFPR	Universidade Federal do Paraná
UEPG	Universidade Estadual de Ponta Grossa

Visual C++ 4.0	Linguagem C visual versão 4.0 da Microsoft para Windows
WINDOWS®	Marca Registrada de Microsoft, Inc.
WINDOWS NT	Sistema operacional para rede da Microsoft



## Lista de Figuras

<b>Figura 1 – Sociedade de Agentes do SISMAT</b>	<b>31</b>
<b>Figura 2 – Tarefas da Sociedade de Agentes do SISMAT</b>	<b>34</b>
<b>Figura 3 – AI solicita tarefa ao AM</b>	<b>40</b>
<b>Figura 4 – AI transmite crença ao AM</b>	<b>40</b>
<b>Figura 5 – AM define local das BDs</b>	<b>42</b>
<b>Figura 6 – AM solicita matrícula ao AMT</b>	<b>42</b>
<b>Figura 7 – AMT valida ra</b>	<b>44</b>
<b>Figura 8 – AMT compõe a matrícula</b>	<b>45</b>
<b>Figura 9 – AMT transmite crença ao AM</b>	<b>45</b>
<b>Figura 10 – AM descarta conhecimentos</b>	<b>49</b>
<b>Figura 11 – Estrutura Registro Acadêmico [UEPG99]</b>	<b>52</b>
<b>Figura 12 – Tela 1: Solicita RA.</b>	<b>53</b>
<b>Figura 13a – Tela 2: Cadastro – Acadêmico s/dependência.</b>	<b>54</b>
<b>Figura 13b – Tela 2: Histórico e Situação Escolar– Acadêmico s/dependência.</b>	<b>54</b>
<b>Figura 14 – Tela 3: Matrícula - Acadêmico s/dependência.</b>	<b>55</b>
<b>Figura 15 – Grade curricular Curso de Informática [UEPG99].</b>	<b>57</b>
<b>Figura 16 – Tela 2: Cadastro e Histórico Escolar – Acadêmico c/dependência.</b>	<b>58</b>
<b>Figura 17 – Tela 3: Matrícula – Acadêmico c/dependência.</b>	<b>59</b>
<b>Figura 18 – Crenças AI</b>	<b>61</b>
<b>Figura 19 – Compromissos AI</b>	<b>62</b>
<b>Figura 20 – Tarefas AI</b>	<b>63</b>
<b>Figura 21 – Crenças AM</b>	<b>64</b>
<b>Figura 22a – Compromissos AM</b>	<b>65</b>
<b>Figura 22b – Compromissos AM</b>	<b>66</b>
<b>Figura 23 – Tarefas AM</b>	<b>67</b>
<b>Figura 24 – Crenças AMT</b>	<b>69</b>
<b>Figura 25 – Compromissos AMT</b>	<b>70</b>
<b>Figura 26 – Tarefas AMT</b>	<b>71</b>

## Lista de Tabelas

<b>Tabela 1 – OOP versus AOP</b>	<b>23</b>
<b>Tabela 2 – Fatos da Sociedade de Agentes do SISMAT</b>	<b>36</b>
<b>Tabela 3 – Ações da Sociedade de Agentes do SISMAT</b>	<b>37</b>
<b>Tabela 4 – Decisões da Sociedade de Agentes do SISMAT</b>	<b>37</b>
<b>Tabela 5 – Estados Mentais Agente Interface</b>	<b>39</b>
<b>Tabela 6 – Estados Mentais Agente Mediador</b>	<b>43</b>
<b>Tabela 7 – Estados Mentais Agente Matriculador</b>	<b>47</b>

## Resumo

Este trabalho apresenta o desenvolvimento do SISMAT - Sistema de Matrícula Inteligente, que tem por objetivo auxiliar professores, coordenadores de curso e pessoal envolvidos na realização de matrículas de acadêmicos de uma universidade. O SISMAT é baseado em um sistema multiagente composto por um agente humano aqui denominado Agente Usuário e três agentes de *software*: Agente Interface, Agente Mediador e Agente Matriculador. O Agente Usuário é aquele que expressa a intenção de efetuar uma matrícula e crê que o SISMAT pode efetua-la. O Agente Interface permite ao Agente Usuário expressar suas intenções ao SISMAT, bem como solicitar e obter conhecimento a respeito de uma matrícula, ou seja o Agente Interface faz a comunicação entre o usuário e a sociedade de agentes. O Agente Mediador gerencia a sociedade de agentes, buscando conhecimento quando necessário e solicitando a execução de tarefas a outro agente quando ele não tem a capacidade de executá-las. O Agente Matriculador, entre outras capacidades, é capaz de compor a matrícula de um acadêmico, através de análises de seu histórico escolar. O SISMAT apresenta características, onde aspectos de interação, cooperação, compromissos, capacidades, crenças, ou seja, características relacionadas ao comportamento social humano, desempenham um papel importante em sistemas baseados em agentes. Aspectos relacionados à concepção, projeto e implementação do SISMAT serão discutidos ao longo deste trabalho. O protótipo foi desenvolvido em LALO (*Langage d'Agents Logiciel Objet*), que utiliza o paradigma de programação orientada a agentes AOP (*Agent-Oriented Programming*).

**Palavras-chave:** Inteligência Artificial Distribuída, Sistemas Multiagente, Agentes Inteligentes, Programação orientada a agentes, Sistemas Baseados em Conhecimento.

## Abstract

This work presents the development of SYSIR - System of Intelligent Registration, which aims to assist teachers, course coordinators and staff involved in the accomplishment of academic registrations at a university. SYSIR is based on a multi-agent system composed of a human agent here denominated User Agent and three software agents: Interface Agent, Mediator Agent and Registrar Agent. The User Agent is that which expresses the intention of making a registration and he believes that SYSIR can perform it. The Interface Agent allows the User Agent to express his intentions to SYSIR, as well as to ask for and obtain knowledge regarding a registration, that is the Interface Agent links the user to the society of agents. The Mediator Agent manages the society of agents, searching for knowledge when necessary and requesting the execution of tasks from another agent when it is not able to perform them. The Registrar Agent, among other capabilities, is able to compose a student's registration through analyses of his/her transcript. SYSIR presents features where aspects of interaction, cooperation, commitments, capabilities, beliefs, that is to say, characteristics related to human social behavior, play an important role in agent-based-systems. Aspects related to the conception, design and implementation of SYSIR will be discussed throughout this work. The prototype was developed in LALO (Langage of Agents Logiciel Objet), which uses the paradigm of agent-oriented programming (AOP).

**Keywords:** Distributed Artificial Intelligence, Multi-Agent Systems, Intelligent Agents, Agent-Oriented Programming, Knowledge-Based Systems.

# 1 Introdução

## 1.1 Contexto do trabalho

Este trabalho apresenta o SISMAT - Sistema de Matrícula Inteligente, cujo objetivo é auxiliar professores, coordenadores de curso e pessoal envolvidos na realização de matrículas de acadêmicos de uma universidade, utilizando agentes inteligentes em seu desenvolvimento.

A utilização de agentes inteligentes na construção de Sistemas Baseados em Conhecimento<sup>1</sup> possibilita o desenvolvimento de diferentes raciocínios e a integração de várias ações para alcançar um determinado objetivo.

O SISMAT é baseado em um Sistema Multiagente (SMA) [DEM92,FER91,WEISS99] composto por um agente humano aqui denominado Agente Usuário e três agentes de *software*: Agente Interface, Agente Mediador e Agente Matriculador. As principais atribuições destes agentes são:

- Agente Usuário (AU) é aquele que expressa a intenção e/ou desejo<sup>2</sup> de efetuar uma matrícula e crê que o SISMAT pode efetua-la.
- Agente Interface (AI) permite ao AU expressar suas intenções ao SISMAT, bem como solicitar e obter conhecimento a respeito de uma matrícula, ou seja o AI faz a comunicação entre o AU e a sociedade de agentes.
- Agente Mediador (AM) gerencia a sociedade de agentes, buscando conhecimento sobre o acadêmico, histórico escolar e regimento interno da universidade, quando necessário, e solicita a execução de tarefas a outro agente quando ele não tem a capacidade de executá-las.
- Agente Matriculador (AMT), entre outras capacidades, é capaz de compor uma matrícula através de análises no histórico escolar do acadêmico.

---

<sup>1</sup> Um sistema baseado em conhecimento é uma realização computacional associada com uma coleção de modelos, cuja modelagem concerne além do conhecimento especialista, várias características de como aquele conhecimento é associado e usado em um ambiente organizacional [SCHB94].

<sup>2</sup> No contexto deste trabalho as intenções e desejos do AU não são tratados como estados mentais deste, e sim como estados intencionais [GIR99] para expressar a sua vontade na elaboração de uma matrícula.

Os agentes dessa arquitetura podem estar em um único computador, ou podem estar localizados em computadores diferentes, estando interligados por uma rede, como por exemplo Internet ou Intranet.

## **1.2 Especificação do problema**

O processo de matrícula em uma universidade geralmente é um problema complexo e trabalhoso. Um dos maiores problemas enfrentados neste processo são os aspectos que envolvem as decisões e os conhecimentos necessários sobre os diversos elementos para a composição da matrícula de um acadêmico.

Dentre estes conhecimentos pode-se destacar: a situação que os alunos apresentam em determinadas fases de sua vida estudantil; as diversas grades curriculares de cada um dos cursos; critérios de avaliação; sistema de frequência; modos de ingresso; codificação de disciplinas, etc., e de uma forma geral tudo sobre o regimento interno da instituição. Estes conhecimentos são necessários para que decisões possam ser tomadas, como por exemplo, determinar em que série e quais disciplinas o aluno deve ser matriculado.

Por estes motivos, o Sistema de Matrícula é um excelente domínio a ser trabalhado com a abordagem SMA. Neste ambiente, a necessidade de um trabalho cooperativo nasce, fundamentalmente, da complexidade e volume de conhecimento necessário. Em um processo de matrícula, nenhum dos indivíduos envolvidos tem todo o conhecimento necessário para individualmente efetuar as matrículas.

## **1.3 Objetivo do Trabalho**

Em um Sistema de Matrícula, equipes de pessoas trabalham em conjunto, sendo cada uma dotada de certo conhecimento, crenças e capacidades próprias correspondentes ao seu curso ou aptidão dentro do sistema e devem interagir entre si para realizar com sucesso a matrícula.

Neste contexto, grande parte do conhecimento é assimilado pela experiência prática [POZO96], onde em cada matrícula (acadêmico) é criado um quadro de associações mentais entre a situação do acadêmico e seu histórico escolar. Neste trabalho propõem-se criar situações em que os agentes do SISMAT criem estas associações e decidam, em que série e quais

disciplinas um acadêmico pode ser matriculado, ou caso contrário, determinem o motivo pelo qual a matrícula não pode ser efetuada.

O objetivo deste trabalho é projetar e implementar um sistema de matrícula inteligente usando como principal fundamentação teórica a abordagem em Inteligência Artificial Distribuída (IAD) e o paradigma SMA.

Através do desenvolvimento do SISMAT, procurou-se atribuir aos seus agentes propriedades [BEL95,FRA96,HEI95] como: autonomia, comunicabilidade, cooperação, iniciativa, reatividade, sociabilidade e reusabilidade. As propriedades que os agentes possuem, definem o comportamento global de um sistema.

Dentre os objetivos gerais pode-se destacar os seguintes objetivos específicos:

- Projetar e implementar um sistema que elabore automaticamente a matrícula de acadêmicos de uma universidade.
- Projetar e implementar um agente interface que permita a comunicação de um agente usuário com os demais agentes do sistema de matrícula.
- Projetar e implementar um agente mediador que gerencie a sociedade de agentes do SISMAT, e que o mesmo tenha a capacidade (entre outras), de localizar e buscar todo o conhecimento necessário para a realização da matrícula de um acadêmico.
- Projetar e implementar um agente matriculador que tenha como principais capacidades: analisar os conhecimentos (situação escolar, cadastro, histórico escolar, matrículas anteriores, etc.) relativos a um acadêmico; com esta análise determinar se este acadêmico pode ser ou não matriculado; e compor a matrícula.
- Estabelecer a comunicação entre os agentes que compõem a sociedade do SISMAT.

O protótipo do sistema foi desenvolvido em LALO (*Langage d'Agents Logiciel Objet*)<sup>3</sup>, que utiliza o paradigma de programação orientada a agentes AOP (*Agent-Oriented Programming*)<sup>4</sup>, e integra conceitos de estados mentais<sup>5</sup> de agentes.

---

<sup>3</sup> O ambiente de desenvolvimento de agentes LALO será abordado no Capítulo 3.

<sup>4</sup> Yoav Shoham propôs um paradigma de programação baseado na noção de agentes deliberativos. Ele chama este paradigma de AOP (*Agent Oriented Programming*). Em AOP um agente é determinado por suas crenças, capacidades, escolhas e compromissos. Todas estas características são chamadas de estados mentais dos agentes [SHO93].

Aspectos relacionados à concepção, projeto e implementação do SISMAT são discutidos ao longo deste trabalho.

## 1.4 Organização do Trabalho

Este trabalho é composto por seis capítulos, nos quais serão abordados os seguintes assuntos:

### **Capítulo 1: Introdução.**

- Constituído por esta seção introdutória, onde faz-se uma descrição resumida do SISMAT, bem como qual é o seu domínio de aplicação e quais são os objetivos a serem alcançados com o desenvolvimento deste trabalho.

### **Capítulo 2: Inteligência Artificial.**

- Abordam-se aspectos relacionados aos princípios teóricos da Inteligência Artificial, bem como algumas definições e a sua evolução para a Inteligência Artificial Distribuída e dentro deste campo a importância e os aspectos que envolvem os Sistemas Multiagentes.

### **Capítulo 3: Agentes.**

- Faz-se uma abordagem sobre agentes inteligentes, destacando as diversas definições propostas por vários autores e a definição adotada neste trabalho. São descritas as propriedades e arquiteturas que os agentes podem possuir, associados a este tema, os comportamentos que os mesmos podem expressar em um Sistema Multiagente. Abordam-se também os estados mentais propostos por Shoham, para aplicação na programação orientada a agentes. Focam-se as principais características de plataformas para o desenvolvimento de agentes e os aspectos técnicos e funcionais do LALO, cuja ferramenta foi utilizada para o desenvolvimento deste trabalho.

---

<sup>5</sup> Os estados mentais de agentes (crenças, capacidades escolhas e compromissos) serão abordados no Capítulo 3 e procura simular de forma restrita o comportamento humano, utilizando uma metáfora dos estados mentais humanos.



#### **Capítulo 4: Arquitetura do SISMAT.**

- Neste capítulo são enfocados os principais aspectos considerados para o projeto e desenvolvimento do protótipo do SISMAT. É feita uma descrição detalhada de cada um dos agentes que compõem a sociedade (Interface, Mediador e Matriculador). Discute-se os estados mentais (crenças, compromissos e capacidades) dos agentes, assim como, os fatos, ações e decisões da sociedade de agentes, necessários a um processo de matrícula.

#### **Capítulo 5: Resultados do SISMAT.**

- Abordam-se neste capítulo os seguintes assuntos: a filosofia de operação do sistema, em que é demonstrada a simplicidade e facilidade oferecida ao usuário; exemplos de situações que podem ocorrer em matrículas de acadêmicos de uma universidade e os respectivos resultados obtidos através do SISMAT; as propriedades atribuídas aos agentes do SISMAT; discute-se a principal vantagem que o paradigma de programação orientada a agentes oferece sobre a programação orientada a objetos, no desenvolvimento de sistemas baseados em conhecimento; e focam-se também, os pontos positivos e negativos da ferramenta de desenvolvimento de agentes LALO.

#### **Capítulo 6: Conclusões.**

- Neste capítulo são descritas as conclusões obtidas no desenvolvimento do SISMAT, são destacadas as propriedades que os agentes propostos neste trabalho apresentam, a importância da modelagem de agentes com estados mentais, a noção de tempo associada aos agentes e as perspectivas de trabalhos futuros.

## Apêndices.

- Em módulo à parte, encontram-se as listagens dos programas escritos para o desenvolvimento do SISMAT, em LALO e C++.
- Relatório Técnico – encontra-se disponível num módulo em separado, um relatório técnico que contempla os aspectos operacionais e as características técnicas da ferramenta de desenvolvimento de agentes LALO (*Langage d'Agents Logiciel Objet*) versão 1.3, cujo trabalho é fruto das pesquisas realizadas durante esta dissertação.

## 2 Inteligência Artificial

### 2.1 Introdução

O desenvolvimento da Inteligência Artificial (IA), especialmente nas últimas décadas, permitiu a constituição das bases utilizadas para compor uma ciência da mente. Através da IA, os cientistas refletem sobre o significado do que é ser inteligente, ter vida mental, ter consciência e outros conceitos empregados nas áreas afins. Se a IA não conseguiu até agora criar um sistema computacional inteligente, ela pelo menos auxiliou na reflexão dos cientistas sobre os processos mentais e seu funcionamento [GIR99].

Pode-se dizer que a IA tem como principal objetivo a representação do comportamento humano através de modelos computacionais. Este comportamento fundamenta-se em dois temas: o conhecimento e a aprendizagem. O conhecimento é sua principal matéria prima, podendo ser representada por diversos formalismos. A aprendizagem também tem despertado muito o interesse da comunidade de inteligência artificial e apresenta-se constantemente em todos os temas relacionados à IA, pois é uma forma de os sistemas evoluírem.

Dentro da comunidade de Inteligência Artificial ainda não foi formulada uma definição que seja satisfatoriamente aceita por todos. Os conceitos encontrados normalmente têm se mostrado incompletos aos objetivos da IA, ou sua conotação não satisfaz totalmente os seus propósitos de estudos. A seguir citam-se algumas definições que confirmam esta situação.

- Para Patrick Winston “Inteligência Artificial é o estudo de conceitos que permitem aos computadores serem inteligentes” [BAR97];
- Para Elaine Rich e Kevin Knight “Inteligência Artificial é o estudo de como fazer os computadores realizarem coisas que, no momento as pessoas fazem melhor” [BAR97];
- Para Michael J. Wooldridge e Nicholas R. Jennings “Inteligência Artificial é um subcampo da ciência da computação que tem por objetivo construir agentes que exibem aspectos de comportamento inteligente” [WOOL96].
- Para Eugene Charniak e Drew McDermott “Inteligência Artificial é o estudo das faculdades mentais com o uso de modelos computacionais” [BAR97].

Com relação às definições anteriormente citadas, percebe-se que a primeira utiliza o termo inteligente, sendo que, se este não for convenientemente definido, torna a definição in-

completa. A segunda definição induz uma competição com os seres humanos, o que dá à IA uma característica de ferramenta com um objetivo limitado. Na terceira, os autores procuram evidenciar a importância da teoria de agentes. A quarta parece a mais completa, pelo fato de buscar aspectos humanos para serem relacionados com modelos computacionais e por estar relacionada a outras áreas de conhecimento, através do termo faculdades mentais, o que torna uma definição mais abrangente.

A IA clássica cuja origem é psicológica, tem seu estudo centrado no comportamento individual do ser humano [GIR99]. No início dos anos 70, iniciaram-se pesquisas acadêmicas em Inteligência Artificial Distribuída (IAD), como uma evolução natural dos Sistemas Distribuídos e da IA clássica. Neste caso, a IAD na sua origem sociológica, utiliza um modelo de inteligência baseado no comportamento social [POZO96].

## 2.2 Inteligência Artificial Distribuída

A Inteligência Artificial Distribuída (IAD) é o estudo do comportamento computacionalmente inteligente, resultante da interação de múltiplas entidades dotadas de certo grau, possivelmente variável, de autonomia. Estas entidades são usualmente chamadas de agentes e o sistema como um todo, sociedade [OLI96].

A IAD difere da área de processamento distribuído por apresentar, além da distribuição dos dados, controle distribuído e envolver uma forte cooperação entre as entidades processadoras.

A área de IAD pode ser classificada de duas maneiras [DEM90]:

- *Inteligência Artificial Distribuída*: está voltada para a solução colaborativa de problemas globais por um grupo distribuído de entidades, que variam de elementos de processamento simples a entidades racionais. A atividade é colaborativa no sentido de que é necessário o compartilhamento de informações e o grupo é distribuído de forma que o controle e os dados são lógicos e algumas vezes geograficamente distribuídos.
- *Inteligência Artificial Descentralizada (IADz)*: está voltada para as atividades de agentes autônomos em um mundo multiagente. Agente é utilizado no sentido amplo para designar uma entidade inteligente, agindo racionalmente com o propósito de atingir seus próprios objetivos e o estado corrente de seu conhecimento. Os agentes podem trabalhar em direção a um único objetivo global, ou rumo a objetivos individuais separados, que podem intera-

gir, sendo então a autonomia dos agentes relacionada com a existência de cada agente independentemente da existência dos demais.

A IAD, em suas duas áreas, trabalha com o conceito de agentes. Esses agentes possuem um papel importantíssimo na resolução dos problemas que a IAD apresenta.

Segundo Bond & Gasser [BOND88], os principais problemas em Inteligência Artificial Distribuída podem ser divididos basicamente em cinco áreas diferentes:

1. Descrição, decomposição e alocação de tarefas - refere-se a como pode ser descrita e decomposta uma tarefa complexa em subtarefas, como essas subtarefas serão alocadas, estática ou dinamicamente e como elas serão realizadas;
2. Interações, linguagens e comunicações - que vocabulário de primitivas um protocolo de comunicação deve apresentar de forma a expressar os conceitos semânticos de um trabalho cooperativo;
3. Coordenação, controle e comportamento coerente - como assegurar um comportamento global coerente do conjunto de agentes, cada agente possuindo seus próprios objetivos e habilidades locais;
4. Conflito e incerteza - como nenhum agente possui a informação completa do ambiente, nascem conflitos e estes devem ser resolvidos. Os dados incompletos e incertos devem ser tratados de forma a garantir resultados coerentes;
5. Linguagem de programação e ambiente - devem ser definidas linguagens de programação que suportem os diferentes requisitos necessários a cada agente pertencente ao sistema.

As linhas de trabalho em IAD são: Resolução Distribuída de Problemas (RDP) e Sociedade de Multiagentes ou Sistemas Multiagentes (SMA). Esta última linha apresenta-se dividida em: Desenvolvimento de Agentes e Estudo de suas *Interfaces* e em Redes Cooperativas de Sistemas Especialistas [DEM90].

Os benefícios da IAD são similares aos benefícios derivados de existência de um grupo de pessoas trabalhando juntas para resolver problemas, que são muito complexos para um só especialista resolver, podendo então, serem resolvidos por um grupo. Os problemas podem ser resolvidos em partes, como pessoas trabalhando em paralelo.

Existem muitas razões para que seja empregado o conceito de Inteligência Artificial Distribuída. Entre elas, pode-se destacar algumas: maior poder de computação, diminuição de

custo de *hardware*, maior segurança em resolução de problemas, confiança em dados, tolerância a falhas, eficiência, velocidade, compartilhamento de recursos, autonomia, etc.

Com a distribuição dos processos, surge o problema da padronização. A linguagem de comunicação entre os processos deve ser padronizada, como também a forma de representar o conhecimento, para que o mesmo seja compartilhado e tratado por processos especialistas distintos sem a perda de informação.

### **2.2.1 Resolução Distribuída de Problemas (RDP)**

Os agentes cooperam uns com os outros, dividindo e compartilhando conhecimento sobre o problema e sobre o processo para obter uma solução. Os agentes são projetados para resolver um problema ou classe de problemas específicos, possuindo, porém, uma visão incompleta do objetivo global [DEM92].

Isto significa que a priori os agentes não podem ser usados para resolver qualquer outro problema similar. Em outras palavras, existe um domínio específico para ação dos agentes. Em RDP, agentes geralmente compartilham um objetivo comum, têm uma linguagem e semântica comuns. Além disso, um único agente nunca é capaz de resolver um problema isoladamente, mas apenas a comunidade de agentes como um todo é capaz de realizar a tarefa.

### **2.2.2 Sistemas Multiagentes (SMA)**

Um ambiente multiagente pode ser definido como um sistema no qual diversos agentes interagem. No paradigma SMA o foco principal consiste em coordenar o comportamento inteligente de um conjunto de agentes autônomos, para obter a solução de um problema apresentado.

Demazeau [DEM92] expõe que sistema multiagente diz respeito às atividades do agente autônomo. Por autônomo, entende-se que o agente tem sua própria existência, independente da existência de qualquer problema a resolver. Os agentes existem em um ambiente comum e cada um pode colaborar com os outros para realizar um objetivo.

Em SMA, os agentes são mais independentes, mas estão competindo pelos mesmos recursos limitados, tais como: tempo, espaço, ferramentas; precisam coordenar-se por razões de eficiência (beneficiando-se de resultados de outros agentes, ou ajudando-os) e devem evitar conflitos. Apenas em situações extremas eles resolvem problemas individualmente.

Mesmo que não utilizem um mesmo vocabulário, eles têm a capacidade de transmitir e mapear seus problemas através de uma forma de representação comum. Estas definições podem parecer conflitantes e na verdade, sem uma análise mais profunda e minuciosa, dúvidas podem ocorrer, pois sistemas de RDP podem ter conceitos de SMA e vice-versa.

Em muitas aplicações, os sistemas sendo formados por módulos independentes, com capacidade de interação entre si, apresentam vantagens significativas sobre os sistemas centralizados. Entre as vantagens destacam-se:

- Modularidade do sistema – é mais fácil criar e manter uma coleção de módulos quase independentes do que um módulo único.
- Eficiência – nem todos os conhecimentos são necessários para todas as tarefas. Quando o sistema é modularizado, tem-se a possibilidade de focalizar os esforços do sistema de solução de problemas do modo mais compensador possível.
- Arquiteturas rápidas – quando os resolvidores de problemas começam a ficar complexos, eles passam a precisar de mais e mais ciclos. Apesar das máquinas estarem cada vez mais rápidas, a real aceleração vem não de um único processador com uma enorme memória associada a ele, mas de um agrupamento de processadores menores, cada um com memória própria. Os sistemas distribuídos têm melhores condições de explorar essas arquiteturas.
- Raciocínio heterogêneo – as técnicas de solução de problemas e os formalismos de representação de conhecimento que mais se prestam para trabalhar em uma parte de um problema podem não ser os melhores para trabalhar em outra parte.
- Múltiplas perspectivas – o conhecimento exigido para solucionar um problema pode não estar na cabeça de um única pessoa. É muito difícil juntar diversas pessoas para criar uma única base de conhecimento coerente e às vezes essa reunião é até mesmo impossível, porque os modelos que cada uma delas tem sobre o domínio em questão são inconsistentes.
- Problemas distribuídos – alguns problemas são inerentemente distribuídos;
- Confiabilidade – se um problema está distribuído entre agentes de sistemas diferentes, a resolução pode continuar mesmo que um dos sistemas apresente falhas.

Contudo, algumas desvantagens podem ser destacadas, principalmente pela independência dos módulos e a necessidade da interação entre eles, que é necessária em um sistema multiagente. As principais são: a complexidade que os sistemas possuem, a comunicação en-

tre os módulos, em função da comunicação, os conflitos que podem surgir e o aspecto cooperação, necessário para que o sistema de fato seja eficiente.

## **2.3 Conclusões**

A finalidade da IA é construir programas funcionais que resolvam problemas das áreas de interesse. Na teoria, não importa como este processo é realizado, qual linguagem é utilizada para alcançar os objetivos ou em que máquina é feito o processamento.

Não há dúvidas de que a capacidade de adaptação a novos ambientes e de resolução de problemas é uma característica importante de sistemas inteligentes.

Sistemas de Inteligência Artificial Distribuída é a classe de sistemas que permite a vários processos autônomos, chamados agentes, realizarem atos de inteligência global, através de processamento local e comunicação interprocessos.

Pode-se afirmar também, que a meta de sistemas baseados na abordagem de IAD é poder coordenar atividades de grupo resolvendo problemas que os sistemas compartilham, total ou parcialmente.

Na próxima seção faz-se uma abordagem aos agentes, visto que são eles as entidades que integram os Sistemas de Inteligência Artificial Distribuída.



## 3 Agentes

### 3.1 Introdução

O termo agente é uma noção central e fundamental para área de [GIR99] Inteligência Artificial Distribuída [OLI96]. Este termo vem sendo utilizado para denotar simples processos de *hardware* e/ou *software* até entidades sofisticadas capazes de realizar tarefas complexas. Esta diversidade reflete o estado atual da área, onde não se possui um consenso do que realmente um agente é [GIR99].

Neste capítulo faz-se uma abordagem sobre agentes inteligentes, destacando as diversas definições propostas por vários autores, e a definição adotada neste trabalho. Focam-se as propriedades e arquiteturas que os agentes podem possuir e os comportamentos que os mesmos podem expressar em um SMA [DEM92]. Neste mesmo contexto são abordados os estados mentais propostos por Shoham [SHO93], e a programação orientada a agentes adotada para o desenvolvimento do SISMAT.

### 3.2 Definições

O surgimento da tecnologia de agentes, trouxe consigo uma variedade de definições sobre o termo *agente*. Em suas propostas, os pesquisadores tentam explicar os seus enfoques, assim como criar novos termos para referenciar seus agentes; alguns destes termos são: agentes inteligentes, interfaces inteligentes, interfaces adaptativas, *knowbots*, *softbots*, *usertbots*, *taskbots*, *personal* agentes e *network* agentes.

Nas definições sobre o conceito de *agente*, observam-se diversas idéias como nas citadas a seguir:

- Um agente é qualquer coisa que pode ser vista como perceptora do ambiente (por sensores) e capaz de atuar nele (por efectores) [RUS95].
- Agente é uma entidade real ou virtual que está imersa em um ambiente onde ele pode realizar ações, que é capaz de perceber e representar parcialmente este ambiente, de comunicar-se com outros agentes e que possui um comportamento autônomo, que é consequência de suas observações, seu conhecimento e interações com outros agentes [FER91].

- Agente é aquele que possui e explicitamente representa, modelos simbólicos do mundo e no qual as decisões são feitas por meio de raciocínio simbólico [WOOL95].
- Agente inteligente é um sistema, situado dentro de um ambiente, que nele percebe e atua continuamente em busca de sua própria agenda, a fim de aplicar o que percebeu em um momento futuro [FRA96].
- Um agente é um programa de computador que funciona em *background* e desenvolve tarefas autônomas conforme as delegadas pelo usuário [RIV96].
- Agentes são programas que travam diálogos, negociam e coordenam transferência de informações [COEN96].
- Os agentes apresentam conceitos de habilidade para execução autônoma e habilidade para executar raciocínio orientado ao domínio [VIR95].
- Agente é uma entidade (computador ou humano) que é capaz de atingir metas e é parte de uma grande comunidade de agentes que tem influência mútua um sobre o outro [HAY99].
- Agente pode ser definido como alguém ou alguma coisa que atua como um representante para outro partido, com o propósito expresso de desempenhar ações que são benéficas para a parte representada. É diferenciado de outras aplicações por suas dimensões e capacidade de interagir independentemente da presença do usuário [HEI95].
- Um agente é um sistema computacional que está situado em algum ambiente e que é capaz de ações autônomas neste, para alcançar seus objetivos de projeto [WEISS99].

Cada autor tenta expressar o que entende a respeito do termo agente, destacando determinadas características pertinentes a sua concepção. Contudo, as principais diferenças são encontradas com respeito ao ambiente onde o agente está imerso e atuará.

O ambiente onde o agente está presente é caracterizado por tudo que o envolve, muitas vezes modelado também como um agente. Verifica-se ainda, que o ambiente como sendo a dispersão do controle, dos dados e do conhecimento pela comunidade de agentes. O ambiente pode ou não incluir outros agentes, e classificado como determinístico, estático ou dinâmico [GIR97].

O mundo do agente pode ser definido como a descrição completa e instantânea do ambiente no qual o agente se encontra [GIR97].

Neste trabalho, a concepção adotada para o termo agente centra-se na seguinte definição: *agentes* são entidades computacionais ou não, que desempenham o papel de executores de tarefas, particularmente ou em conjunto, para resolver problemas pertinentes a sistemas

baseados em conhecimento, i.e. aplicando conhecimento, atitudes sociais e mentais sobre o propósito da ação.

Destacam-se nesta definição, dois aspectos principais: o fato do agente não ser necessariamente uma entidade computacional; e sua essência estar relacionada à aplicação de conhecimento na execução de tarefas, utilizando-se inclusive de propriedades sociais e mentais (desejos, crenças, convicções, sociabilidade, comunicabilidade, etc.) inerentes ao ser humano.

Um conjunto de Agentes Inteligentes pode caracterizar a existência de um SMA [DEM92,FER91], quando eles atuam como uma sociedade, interagindo entre si, para que o objetivo comum seja alcançado.

Um agente deve apresentar algumas propriedades. Elas definem qual é a classificação do agente, por exemplo, um agente inteligente, reativo, capaz de aprender, etc. A seguir, serão destacadas propriedades relevantes, de acordo com a literatura pesquisada.

### 3.3 Propriedades de Agentes

Propriedades de agentes são características de um sistema e têm um efeito no comportamento global do sistema. Para que o sistema seja considerado um agente, ele não necessita apresentar todas as propriedades, mas algumas delas são necessárias.

O ambiente e seu respectivo projeto (*design*) irá determinar as propriedades que o agente terá [GIR99]. A seguir, apresentam-se algumas destas propriedades, fundamentadas em [BEL95,FRA96,HEI95,DEM92].

- **Autonomia**

“Autonomia” é a capacidade que o agente pode possuir, para decidir sobre a resolução de determinada tarefa, independentemente das intenções e desejos de outros agentes, ou ainda, segundo Demazeau [DEM92], por autônomo, entende-se que o agente tem sua própria existência, independente da existência de qualquer problema a resolver.

- **Mobilidade**

“Mobilidade” é a capacidade que os agentes podem possuir de mover-se espacialmente buscando ou conduzindo conhecimento, com o objetivo de resolver determinadas tarefas.

- **Comunicabilidade**

“Comunicabilidade” é a capacidade que os agentes podem possuir de trocar conhecimento entre si, para alcançar os objetivos propostos por suas tarefas.

- **Aprendizagem**

“Aprendizagem” é a capacidade que o agente pode possuir para assimilar conhecimento do ambiente e de outros agentes, com o objetivo de auxiliar na execução das tarefas, para as quais o mesmo foi concebido.

- **Reatividade**

“Reatividade” é o conhecimento que os agentes podem possuir para reagir, adaptar-se e responder aos estímulos oriundos do ambiente ou de outros agentes.

- **Iniciativa**

“Iniciativa” é a habilidade de exibir comportamento direcionado ao objetivo, oportunisto e que não reage simplesmente ao seu ambiente.

- **Sociabilidade**

“Sociabilidade” entre agentes significa imitar certas características inteligentes dos humanos, com o objetivo de atribuir maior grau de inteligência (capacidade de manipular conhecimento) para os agentes, onde estes demonstram conhecimento e segurança para resolução das tarefas para as quais são concebidos.

- **Percepção**

Um agente deve perceber o ambiente e as mudanças ocorridas neste. É o objetivo de adaptar-se a novas situações, buscando inclusive novos conhecimentos, caso estejam disponíveis.

- **Cooperação**

É a propriedade relacionada à sociabilidade, pois normalmente em um ambiente social deve existir o espírito de cooperação. Em termos de agentes, o conjunto de seus conhecimentos, associados às capacidades de executar determinadas tarefas, permite-se afirmar que, em um ambiente cooperativo é possível obter maior poder de resolução de problemas.

Através das propriedades que os agentes podem exibir, é possível obter algumas semelhanças relacionadas ao comportamento de seres humanos. Esta semelhança abrange aspectos sociais e principalmente aqueles que envolvem o compartilhamento de capacidades e conhecimentos inerentes a cada indivíduo. Neste contexto, pode-se traçar um paralelo com características humanas, nas quais cada indivíduo possui seus próprios conhecimentos e capacidades, e por estar inserido em um ambiente social, necessita compartilhar seus conhecimentos e capacidades para alcançar seus objetivos.

### **3.4 Compartilhamento de conhecimento entre os agentes**

No contexto das propriedades sociais dos agentes, que envolvem cooperação, surge o problema da comunicação e o compartilhamento de conhecimento entre os agentes. Esta situação, abrange alguns aspectos relacionados à interface.

Segundo Brazdil [BRA91], nas interfaces entre os agentes, é necessário decidir que tipo de declaração os agentes serão capazes de gerar e compreender. Obviamente, o projeto da interface é cuidadosamente relatado para o projeto da arquitetura de todo o sistema. Na comunicação, um termo pode ter diferentes significados para diferentes agentes, o que é chamado de conflito [HUB98]. Diferentes termos podem, entretanto, ter significado similar, o que significa uma correspondência.

Neste contexto, aparece o aspecto da especificação formal utilizada para a comunicação e o compartilhamento de conhecimento entre os agentes. Considerando estes aspectos, agentes podem fazer uso de ontologias e assim compartilhar e reusar conhecimento entre eles. Entretanto, os agentes podem compartilhar um vocabulário e necessariamente não compartilhar uma mesma base de conhecimento.

Isto quer dizer que uma ontologia [WEISS99] é uma descrição (como uma especificação formal de um programa) dos conceitos e relações que podem existir para um agente ou uma comunidade de agentes. Em outras palavras uma ontologia comum pode definir o vocabulário com o qual perguntas e afirmações são trocadas entre os agentes. Por exemplo, as palavras em um planejamento ontológico podem ser termos técnicos, que coordenam a interpretação das entradas e saídas.

Para Gruber [GRU93], ontologias são acordos sobre conceitualizações compartilhadas. Conceitualizações compartilhadas incluem *frameworks* conceituais para modelagem do conhecimento do domínio: protocolos específicos de conteúdo para comunicação entre agentes de interação e acordos sobre a representação de teorias de domínio particulares.

Percebe-se que as ontologias representam um papel importante em sistemas baseados em conhecimento e que elas poderão ser projetadas de tal modo, que serão minimizados os efeitos do problema de interação.

Uma das possíveis conquistas poderia ser, inclusive, a criação de bibliotecas de ontologias mais amplas com uma abrangência maior de termos de conhecimento, mas isto ainda não foi conseguido satisfatoriamente, em virtude da amplitude de termos necessários e homônimos que podem ocorrer em muitos casos.

Além do aspecto da especificação formal utilizada para a comunicação e compartilhamento de conhecimento, outros passos também devem ser seguidos para desenvolver e caracterizar um agente. O primeiro é uma análise minuciosa do ambiente e das tarefas para determinar os requerimentos da conduta dos agentes. Isto significa a concepção do modelo de agente.

Os modelos de agentes devem implementar explicitamente as características desejadas e a arquitetura de um agente será o resultado do projeto e implementação de todas as propriedades e habilidades que um determinado agente apresentará [POZO96].

### 3.5 Arquitetura de um agente

A arquitetura de um agente determina uma metodologia para a construção de agentes. Ela especifica como o agente pode ser decomposto na construção de um conjunto de módulos componentes e como esses módulos devem interagir. O conjunto total de módulos e suas interações devem fornecer uma resposta de como a percepção do ambiente e o estado interno do agente determinam suas ações. Uma arquitetura abrange técnicas e algoritmos que suportam essa metodologia [MAES91].

Para que uma tarefa ou projeto sejam adequados à aplicação da abordagem de agentes, é necessário identificar algumas características comuns dos agentes, relacionando-as com as perspectivas da resolução da tarefa proposta. Andrew Wood [WOOD94] especifica as seguintes propriedades: adaptação, resolução de problemas, explanação, guia, autonomia e assincronia.

Contudo, em agentes cooperativos outras características também são importantes, como relações estáticas e dinâmicas entre atitudes mentais [BRZ97]. Brazier destaca que um agente precisa ser capaz de raciocinar sobre seu próprio processo, suas próprias tarefas, outros agentes e o mundo. Em outras palavras, um agente precisa ser capaz de realizar quatro tarefas:

1. Controlar seu próprio processo;
2. Executar suas próprias tarefas específicas;
3. Administrar seu conhecimento, de outros agentes e sua comunicação com estes agentes;
4. Administrar seu conhecimento e interação com o mundo.

Observa-se que agentes autônomos, normalmente, exibem uma rica variedade de comportamento reflectivo, eles raciocinam sobre vários aspectos: seu próprio comportamento, sobre os comportamentos de outros agentes, o ambiente e interações entre eles.

Estas propriedades disponíveis nos agentes estão diretamente relacionadas à tarefa ou aplicação, para a qual o agente é proposto. O resultado espelha-se em sua arquitetura, o que dá condições ao projetista de adaptar e modelar o seu agente às suas necessidades.

A arquitetura de um agente mostra como ele está implementado em relação às suas propriedades, sua estrutura e como os módulos que o compõem podem interagir, garantindo sua funcionalidade [MAES91]. De uma forma geral, a arquitetura de um agente especifica sua estrutura e comportamento.

Conforme Wooldridge, as arquiteturas de agentes podem ser divididas em três tipos: Arquitetura Cognitiva ou Deliberativa, Arquitetura Reativa e Arquitetura Híbrida [WOOL95].

Na arquitetura cognitiva ou deliberativa, os agentes possuem uma representação explícita do mundo e dos membros da sociedade. Eles podem raciocinar sobre as ações tomadas no passado e planejar as ações a serem tomadas no futuro. Os agentes cognitivos baseiam-se em mecanismos de processamento simbólico existentes nos sistemas mais tradicionais da Inteligência Artificial, como redes semânticas, sistemas de regras, etc.

A arquitetura reativa não inclui nenhum tipo de modelo simbólico do mundo e não utiliza raciocínio simbólico complexo, somente reage às ações que ocorrem no ambiente.

A arquitetura híbrida mistura componentes das arquiteturas cognitivas e reativas com o objetivo de torná-la mais adequada e funcional para a construção de agentes.

Yoav Shoham [SHO93] propôs um paradigma de programação baseado na noção de agentes deliberativos. Ele chama este paradigma de AOP (*Agent Oriented Programming*). Em AOP um agente é determinado por suas crenças, capacidades e compromissos.

Todas estas características são chamadas de estados mentais dos agentes. AOP pode ser visto como uma especialização de OOP (*Object-Oriented Programming*) onde agentes são objetos que possuem estados mentais (convicções, capacidades e compromissos) e a noção de tempo.

Diferentemente de um objeto que reage sempre do mesmo modo para uma mesma mensagem, um agente tem seu próprio modo de tratá-la e pode agir de modos diferentes para a mesma mensagem, de acordo com o seu estado mental, o qual deve ter uma correspondência aos estados mentais humanos, embora essa correspondência necessariamente não seja exata [GIR99].

Do ponto de vista de um agente, significa que uma mesma ação executada duas vezes em circunstâncias aparentemente idênticas, pode parecer ter efeitos completamente diferentes e em situações particulares pode falhar para ter um efeito desejado [WEISS99]. Na tabela 1 descrevem-se as diferenças definidas por Shoham [SHO93], entre os paradigmas OOP e AOP.

A seguir, apresentam-se os estados mentais propostos por Shoham, de acordo com o paradigma de programação orientada a agentes AOP.



- OOP versus AOP -		
	OOP	AOP
Unidade básica	Objeto	Agente
Estado de definição de parâmetros da unidade básica	Sem restrições	Crenças, compromissos, capacidades, escolhas, ....
Processo de computação	Métodos de resposta e passagem de mensagens.	Métodos de resposta e passagem de mensagens.
Tipos de mensagens	Sem restrições	Informações, pedidos, ofertas, promessas, recusa, ...
Restrições sobre os métodos.	Nenhum	Honestidade, consistência, ...

Tabela 1 – OOP versus AOP

### 3.6 Estados Mentais

Para Shoham o termo agente é uma entidade cujo estado consiste de componentes mentais tais como crenças, capacidades, escolhas e compromissos. Estes componentes mentais referem-se aos estados mentais internos dos agentes, através dos quais, os agentes se relacionam com o ambiente e com outros agentes com os quais interagem.

Shoham propõe a seguinte visão informal do mundo, para aplicação da programação orientada a agentes:

- A qualquer instante, o futuro é determinado por dois fatores: o passado histórico e as ações correntes dos agentes. Por exemplo: só o passado histórico não determina se um robô movimentou seu braço; isto é determinado se realmente o robô realiza a ação apropriada.
- As ações de um agente são determinadas pelas suas *decisões* ou *escolhas*, ou seja, os fatos são verdadeiros por razões naturais ou porque os agentes decidem realizá-los;
- As decisões são logicamente restritas, embora não determinadas pelas *crenças* do agente; essas *crenças* referem-se ao estado do mundo (passado, presente ou futuro), ao estado mental de outros agentes e as suas *capacidades* e de outros agentes. Por exemplo: se um robô acredita que é incapaz de passar por uma porta estreita, ele decide não passar por ela.
- As decisões também são restritas por decisões prévias. Por exemplo: um robô não pode decidir estar na sala 5 em cinco minutos, se ele já decidiu estar na sala 3 no mesmo momento.

Essas perspectivas motivaram Shoham a introduzir dois estados mentais básicos: *crença* e *decisão* (ou escolha) e um terceiro estado, *capacidade*, o qual não é um construtor mental, pois não define os atributos do estado mental de um agente. Essas são as categorias mentais adotadas pelo autor, porém ao invés de *escolha* como básico, introduz a noção de *obrigação* ou *compromisso*, e trata *decisão* simplesmente como uma obrigação do próprio agente. Associados a estes componentes mentais são considerados o *tempo* e a *ação*.

Shoham reconhece que restringindo a apenas esses estados mentais, exclui a representação de *motivação* e não assume que os agentes sejam racionais, mas que suas crenças, obrigações e capacidades são internamente e mutuamente consistentes, embora esteja convencido de que no futuro deverá considerar noções mais fortes de racionalidade.

Descreve-se a seguir, a linguagem para os estados mentais e seus componentes conforme a abordagem de Shoham [SHO93].

- *Tempo*

Tempo é básico para categorias mentais, onde se acredita em coisas em diferentes momentos e diferentes vezes. A representação adotada é:

`holding(robot, cup)t`

onde, o robô apanha a xícara no tempo *t*.

- *Ação*

As ações podem ocorrer em diferentes momentos, e, dependendo das circunstâncias, em um determinado instante podem ter um certo efeito. Para o propósito atual, Shoham não distingue ações e fatos, e a ocorrência de uma ação seria representada por um fato correspondente.

Por exemplo, ao invés de dizer que o robô executa a ação *raise-arm* em um tempo *t*, é dito que a sentença:

`raise-arm(robot)t`

é verdadeira.

- *Crença*

Crença é representada por um operador modal B, como:

$$B_a^t \varphi$$

significando que o agente **a**, no tempo **t**, acredita na sentença  $\varphi$ .

- *Obrigação*

Representada pelo operador modal OBL, na forma,

$$OBL_{a,b}^t \varphi$$

significa, no tempo **t**, o agente **a** está obrigado ou comprometido com o agente **b** sobre  $\varphi$ .

As ações são representadas simplesmente como fatos, e o agente está obrigado a manter o fato em vez de executar a ação.

- *Decisão*

Segundo Shoham, a liberdade de escolha entre diversas ações é fundamental para a noção de agentes e normalmente se considera *decisão* ou *escolha* como um noção primitiva. A definição de *obrigação* oferece uma alternativa:

Definir *decisão* é como obrigar ou comprometer a si próprio, onde:

$$DEC_a^t \varphi =_{\text{def}} OBL_{a,a}^t \varphi$$

O termo *escolha* é utilizado no sentido de decisão; um agente escolheu algo se ele decidiu que aquilo é verdade. Por exemplo: muitos podem decidir ter seu próprio par de sapatos, mas poucos podem decidir ter seu próprio iate.

- *Capacidade:*

Um agente não decidirá fazer algo se ele acredita não ser capaz. Existe um debate a respeito de se capacidade é melhor definida em termos mentais ou não. Pode-se defini-la dizendo que o agente **a** é *capaz* de  $\varphi$  se é verdadeira a seguinte sentença: se **a** decidir  $\varphi$ , então  $\varphi$  será verdade.

Shoham não entra no mérito da questão levantada e simplesmente introduz a noção de capacidade como  $CAN_a^t \varphi$  para representar o fato que, no tempo  $t$ , o agente  $a$  é capaz de  $\varphi$  e também define *apto* a partir de *capacidade*, como:

$$ABLE_a \varphi =_{\text{def}} CAN_a^{\text{time}(\varphi)} \varphi$$

Onde  $\text{time}(\varphi)$  é o maior tempo que aparece na sentença  $\varphi$ .

Por exemplo: no tempo 5 o robô pode ser capaz de assegurar-se que a porta é aberta no tempo 8, mas no tempo 6 ele poderia não ter aquela capacidade:

$$ABLE_{\text{robot}} \text{open}(\text{door})^5 = CAN_{\text{robot}}^5 \text{open}(\text{door})^5$$

Esta é a representação dos estados mentais propostos por Shoham de acordo com o paradigma de programação orientada a agentes AOP. Em [GIR97],[GIR99] pode-se encontrar um rico material a respeito dos estados mentais propostos por Shoham, assim como análises de componentes mentais propostos por outros autores [BRA89], [COR94], [HAL92], [WOOL92]. A seguir, apresentam-se características de ambientes de desenvolvimento de agentes, bem como a plataforma de desenvolvimento que foi utilizada no desenvolvimento deste trabalho.

### 3.7 Ambientes de desenvolvimento de Agentes

Como já abordado, os agentes inteligentes apresentam características que os diferenciam de sistemas convencionais, portanto ferramentas mais específicas para o seu desenvolvimento são necessárias.

Como em toda teoria dos sistemas ligados à Área da Computação e dos Sistemas Inteligentes, a Teoria dos Agentes precisa de um conjunto de ferramentas para poder ser aplicada na prática. Este conjunto de ferramentas é constituído pelas Linguagens de Desenvolvimento de Agentes e suas Bibliotecas, que servem de ponte entre as idéias expressas nas teorias e os programas que efetivamente serão executados nos computadores [PIER96].

A teoria de Agentes Inteligentes ainda é relativamente recente, existem ainda muitos exemplos de aplicações acadêmicas e poucos de uso comercial [LESS99], apesar de existirem sistemas em uso, a maioria ainda é fruto de parcerias entre os centros de pesquisa das universidades e as empresas, principalmente estatais.

A maioria das abordagens utilizadas nas ferramentas de desenvolvimento de agentes é eminentemente proprietária. A sua utilização em larga escala, gera um alto custo e muitas vezes ainda injustificável.

### 3.7.1 Diferenças entre ambientes de desenvolvimento de Agentes

Quando pretende-se realizar alguma classificação ou diferenciação entre as diferentes alternativas para a criação de agentes inteligentes distribuídos deve-se levar em consideração os seguintes fatores [PIER96]:

1. Linguagens Genéricas: trata-se do desenvolvimento dos sistemas de agentes partindo-se praticamente do zero, somente com os recursos oferecidos pelas linguagens comerciais (com suas bibliotecas padrões) normalmente utilizadas para este fim, como C, C++, *Pascal*, *Algol*, *Prolog*, *Lisp*, etc, ou seja, implementando todas as características da teoria de agentes através da codificação e teste de cada uma das rotinas;
2. Frameworks: neste caso o desenvolvedor do sistema normalmente adquire também uma biblioteca específica para a atividade, que cobre os pontos principais, os mais complexos e gerais da teoria, ficando para a equipe de desenvolvimento a modelagem do agente visando as tarefas que o mesmo deverá executar;
3. Linguagens Específicas: nesta opção a parte relativa ao agente está pronta, sob a forma de um esqueleto de aplicação, que atende a um conjunto específico de áreas de atuação de agentes distribuídos, como: aprendizagem, pesquisa, processamento de tarefas, etc. Nesta abordagem, a modelagem do agente prende-se mais a adaptação do mesmo ao ambiente distribuído em que irá atuar e à definição específica da finalidade do mesmo.

### 3.7.2 Exemplos de Plataformas de desenvolvimento de agentes

Na busca de plataformas de desenvolvimento de agentes para a implementação do SISMAT, efetuou-se uma pesquisa para detectar quais são os ambientes disponíveis atualmente. Nesta pesquisa dois grupos foram distintos: ferramentas de desenvolvimento de agentes comerciais e as de projetos de pesquisa e acadêmicos.

Ao efetuar-se a seleção, os ambientes comerciais tiveram que ser descartados por questões econômicas. A seleção concentrou-se nas plataformas com origem em projetos de pesquisa e acadêmicos. Ao selecionar-se a ferramenta, além da flexibilidade que a mesma oferece, o critério de seleção centrou-se principalmente em dois aspectos: que a ferramenta permitisse trabalhar com agentes cognitivos e com estados mentais. Nesta seleção, optou-se pelo LALO que enquadrava-se no critério adotado.

Na seção seguinte, uma descrição sobre o LALO, cuja ferramenta, como mencionado anteriormente foi utilizada no desenvolvimento do SISMAT.

### 3.7.3 LALO (Langage d'Agents Logiciel Objet)

LALO é uma linguagem de Programação Orientada a Agentes e um *framework* para desenvolvimento de sistemas multiagentes inteligentes. Desenvolvido por Daniel Gauvin, Hervé Marchal, Rober Gauthier, Jean-François Rizand e Vince Delle Donne pertencentes ao CRIM - Centre de Recherche Informatique de Montreal [LALO97].

A arquitetura proposta é extensível e oferece a possibilidade da criação de sistemas multiagentes incluindo agentes reativos bem como deliberativos.

Implementado em C++, oferece um conjunto de classes para a construção dos agentes. O programador pode ampliar o conjunto existente adicionando novas subclasses. A linguagem de comunicação utilizada é o KQML<sup>6</sup> (*Knowledge Query and Manipulation Language*).

---

<sup>6</sup> KQML - é uma linguagem projetada para suportar interações entre agentes inteligentes. Ela foi desenvolvida por um grupo de pesquisadores interessados em sistemas de comunicações entre agentes do Knowledge-Sharing Effort (KSE), patrocinado pelo Advanced Research Projects Agency (ARPA) [LAB97.HUB98].

LALO possui uma linguagem própria e seu compilador, permitindo o desenvolvedor programar seus agentes no paradigma AOP. Os programas gerados pelo compilador LALO geram fontes em C++ que podem ser compilados pelos compiladores relativos às arquiteturas de hardware suportadas (Windows NT 4 e Windows 95 – usando Visual C++ 4.0, UNIX Solaris 2.5 em plataforma Sun – usando G++ (gnu) versão 2.7.2, Hewlett Packard HP-UX 9.0.1 – usando G++ (gnu) versão 2.6, SGI IRIX5.3 – usando G++ (gnu) 2.6).

As principais razões que motivaram a escolha do LALO para o desenvolvimento do SISMAT são as seguintes:

1. Permitir a programação orientada a agentes;
2. Permitir o desenvolvimento de sistemas multiagentes;
3. Permitir o desenvolvimento de agentes com comportamentos reativo e cognitivo;
4. Possuir uma linguagem de programação própria;
5. Possuir como linguagem de comunicação entre os agentes, o KQML;
6. Possuir uma documentação satisfatória;

### 3.8 Conclusões

Conclui-se que desenvolver sistemas baseados em conhecimento, utilizando agentes modelados com estados mentais, atribui a estes sistemas, características especiais. Mais especificamente, o paradigma AOP permite aos agentes, responderem as mesmas mensagens de modos diferentes de acordo com seu estado mental, e também a noção de tempo que possibilita aos mesmos, tomarem decisões não somente com base em seu estado mental atual, mas também em relação ao passado e futuro.

Diante desta abordagem sobre os agentes e suas propriedades, é possível perceber que agentes inteligentes são de fundamental importância para o desenvolvimento de sistemas baseados em conhecimento.

As habilidades que os agentes possuem e as propriedades atribuídas a eles, em sua semelhança com atitudes dos humanos, podem tornar os computadores ferramentas cada vez mais poderosas. Podendo ampliar a gama de problemas que podem ser resolvidos e criam condições favoráveis à melhoria e eficácia de problemas já resolvidos.

A seguir, discute-se como os conceitos abordados anteriormente foram utilizados na concepção, projeto e implementação do SISMAT.

## 4 SISMAT

### 4.1 Introdução

O protótipo do SISMAT foi desenvolvido no ambiente de desenvolvimento de agentes LALO e a Universidade Estadual de Ponta Grossa foi usada como modelo (critérios de matrícula adotados para a graduação) [UEPG99].

Uma matrícula constitui-se da formulação periódica, de um plano de estudos a ser cumprido pelo acadêmico, dos períodos letivos em que estiver realizando o curso, conforme o regime interno da instituição e dos calendários estabelecidos. Ao ingressar na UEPG, o acadêmico recebe um código numérico que será a sua identificação. Este código é chamado de Registro Acadêmico (RA) [UEPG99].

Durante o curso e de acordo com o seu rendimento escolar, um acadêmico pode possuir várias situações escolares, podendo ser as seguintes: calouro, aprovado “sem dependência”<sup>7</sup>, aprovado “com dependência”<sup>8</sup> (reprovado até 2 disciplinas), retido - reprovado em mais de duas disciplinas, retido - reprovado em dependência, trancado, formado, jubilado, transferido e reabertura de curso.

Neste trabalho, será considerado um processo normal de matrícula, quando o acadêmico tem um histórico escolar regular e todos os conhecimentos sobre o seu cadastro, histórico escolar, grade curricular são encontrados. Na definição de sua situação escolar estará aprovado em todas as disciplinas sem dependências e poderá ser matriculado na série seguinte a que se encontrar.

A realização de uma matrícula exige um trabalho cooperativo, devido à complexidade e volume de conhecimento necessário. Neste contexto, nenhum dos elementos envolvidos possui todo o conhecimento necessário para individualmente efetuar uma matrícula.

Desta forma o SISMAT (figura 1) é composto por um agente humano aqui denominado Agente Usuário (AU) e três agentes de *software*: Agente Interface (AI), Agente Mediador (AM) e Agente Matriculador (AMT).

O SISMAT é implementado com agentes reativos e cognitivos, pois o AI é dotado de comportamento reativo com relação aos eventos que ocorrem em seu ambiente, como por

---

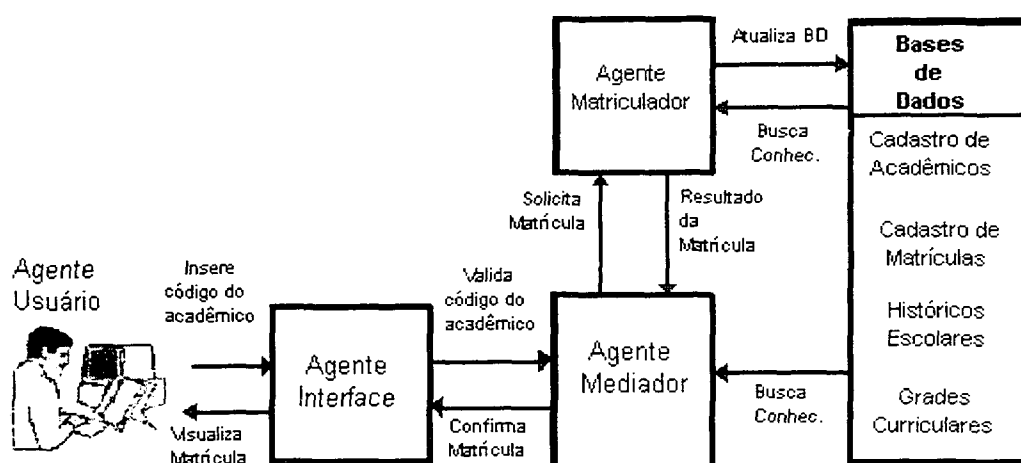
<sup>7</sup> Acadêmico com situação regular sem reprovações.

<sup>8</sup> Será descrito em detalhes no próximo capítulo.



exemplo quando o AU solicita a execução de uma tarefa. O AM e o AMT possuem comportamento cognitivo, pois existe uma definição simbólica do mundo para a tomada de decisões e também possuem a capacidade de raciocinarem sobre suas ações anteriores através de sua Base de Conhecimento (BC) [LALO97].

Cada um dos agentes dessa arquitetura possui: um conjunto de tarefas (capacidades) específicas que deve realizar; um conjunto de compromissos que deve cumprir em relação a outros agentes da sociedade; e um conjunto de crenças em determinados fatos, sobre os quais deve tomar decisões, dependendo de seu estado mental (crenças, compromissos, capacidades).



**Figura 1 – Sociedade de Agentes do SISMAT**

## 4.2 Tarefas da sociedade de agentes do SISMAT

Na figura 2 estão descritas as tarefas básicas de cada um dos agentes que compõem a sociedade do SISMAT. Para facilitar a compreensão da figura, considerou-se, como exemplo de ordem de execução das tarefas, um processo normal de matrícula. O motivo desta escolha justifica-se pela complexidade do problema, pois inúmeras situações podem ocorrer, o que pode alterar a ordem e/ou fazer com que determinadas tarefas não sejam executadas, por exemplo: caso o RA inserido pelo AU seja inválido, o AM retornaria esta crença ao AI, que por sua vez exibiria uma mensagem ao AU, para que o mesmo entre com outro RA. Portanto, neste caso o AM executaria apenas a tarefa de validação do RA e a tarefa de descarte de conhecimento<sup>9</sup>.

<sup>9</sup> Esta tarefa será discutida em detalhes no final deste capítulo.

A divisão das tarefas está de conformidade com as competências<sup>10</sup> de cada um dos agentes que compõem esta sociedade. Neste contexto, pode-se destacar os seguintes aspectos:

- O AI, tem como principal objetivo ser a interface entre o usuário e o SISMAT, portanto suas principais tarefas são: receber os conhecimentos do AU e transmiti-los ao AM;
- O AM tem como finalidade principal gerenciar a sociedade, desta forma suas tarefas visam principalmente a busca e validação dos conhecimentos e a exibição destes ao AU. Através das capacidades do AM, agiliza-se os processos de matrícula, principalmente por duas situações que podem ocorrer durante a utilização do SISMAT:

- 1- O AM, através das análises dos conhecimentos obtidos nas BDs, pode impedir que uma matrícula indevida seja solicitada ao AMT, evitando um gasto de tempo e processamento desnecessários, por exemplo: quando o AM através do RA, verifica que um acadêmico já está matriculado em outro curso<sup>11</sup> (condição que impede um acadêmico de ser matriculado);
- 2- No estudos realizados sobre os procedimentos de matrícula, percebeu-se que, em muitos casos, a intenção do AU é apenas fazer uma consulta no cadastro do acadêmico e/ou histórico escolar. A situação é resolvida pela exibição que o AM faz ao AU do cadastro e histórico escolar do acadêmico;

Note-se que nas duas situações abordadas anteriormente, o AMT não é solicitado para executar nenhuma tarefa.

- O AMT, através de suas tarefas, tem como principal capacidade, efetuar a matrícula de um acadêmico. Pode-se dizer que o objetivo a ser alcançado pelo AMT, é coletivo, pois a realização das matrículas é meta (implícita) a ser alcançada pela sociedade de agentes do SISMAT como um todo. Além da realização da matrícula, o AMT possui entre outras, uma tarefa importante que é a atualização das BDs. Esta tarefa possui uma característica especial no contexto de uma matrícula, pois um novo histórico escolar oficial é definido para o acadêmico a partir daquele momento.

A decisão dos agentes de “qual” e/ou “quais” tarefas devem ser executadas, está diretamente relacionada aos seus estados mentais. Os estados mentais envolvidos neste processo de decisão, são as crenças e os compromissos, por exemplo: quando o AM possui a *crença* de

---

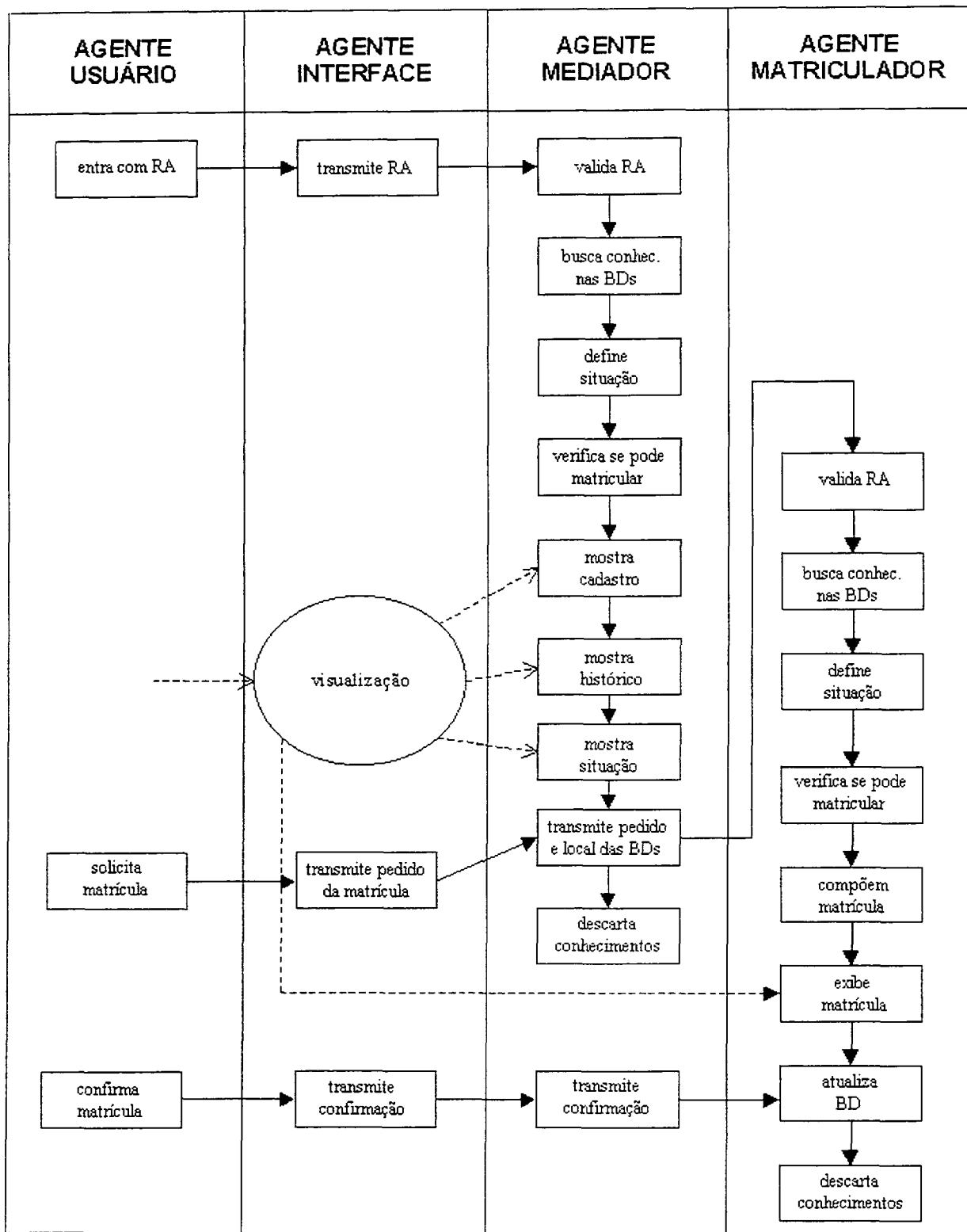
<sup>10</sup> O autor introduz este termo, como sendo as capacidades (aptidões, habilidades) implícitas do agente para realizar determinadas tarefas, a um nível geral da sociedade, por exemplo: o AM tem a capacidade de gerenciar a sociedade do SISMAT; o AMT tem a capacidade de elaborar uma matrícula; o AI tem a capacidade de comunicar-se com o AU.

que o AU tem a intenção de matricular o acadêmico, toma a *decisão* de cumprir um *compromisso* com o AI que é o de executar a *tarefa* que define o local das BDs relativas ao acadêmico.

Porém, para que os agentes da sociedade do SISMAT possam atuar de acordo com seus estados mentais, inúmeros fatos, ações e decisões devem ser considerados, sobre os quais os agentes formarão suas crenças, assumirão compromissos e aplicarão suas capacidades na realização deste objetivo.

---

<sup>11</sup> Aos acadêmicos da UEPG é vedada a matrícula simultânea em mais de um curso de graduação [UEPG99].



RA - Registro Acadêmico | BD - Base de Dados

Figura 2 – Tarefas da Sociedade de Agentes do SISMAT

### 4.3 Estados Mentais - SISMAT

Na fase de modelagem do SISMAT, para que se pudesse conhecer quais estados mentais estariam envolvidos em um processo de matrícula, foi necessário definir sobre quais ocorrências eles seriam aplicados. Nas tabelas 2,3 e 4 relacionam-se os principais fatos, ações e decisões considerados na modelagem da sociedade de agentes do SISMAT, que foram obtidos através de uma análise minuciosa em todo o processo de matrícula.

Através da definição destes fatos, ações e decisões globais, foi possível fazer um refinamento do sistema de matrícula, e obter a associação dos estados mentais com seus respectivos agentes, visando sua implementação.

A fase de refinamento do sistema de matrícula constituiu-se da atribuição de competências (capacidades) a cada um dos agentes, ou seja, como estes fatos, ações e decisões foram definidos globalmente. Eles devem ser associados individualmente aos agentes que compõem a sociedade. Por exemplo: considere-se o fato “Existência da BD em um determinado local”, no SISMAT este fato foi associado ao AM, pelo motivo de que uma das suas principais competências é ser um mediador no sistema. Sendo um mediador, cabe a este agente buscar e validar conhecimentos. A crença do AM sobre este fato, pode ser verdadeira ou falsa, em um determinado momento. Em um processamento, caso o AM encontre a BD relativa ao acadêmico, sua crença sobre este fato será verdadeira. Com esta *crença* verdadeira, o AM pode tomar uma *decisão* de “Se o acadêmico existe, transferir os dados cadastrais para o *site* do agente”. Ao executar esta decisão, o AM estará executando a *ação* “Acessar dados cadastrais do acadêmico”, ou seja, desempenhando uma de suas capacidades.

Deste modo, as *crenças* consistem de *fatos*, em que os agentes do SISMAT poderão acreditar, como sendo verdadeiros ou falsos em um determinado momento. De acordo com o resultado de uma crença, os agentes tomarão as suas *decisões*, para determinar quais *tarefas* (*ações*) deverão ser executadas.

FATOS
<ul style="list-style-type: none"> <li>✓ Existência da BD em um determinado local;</li> <li>✓ Existência do cadastro do acadêmico na BD;</li> <li>✓ Existência da grade curricular (fluxograma) do curso na BD;</li> <li>✓ Existência do histórico escolar do acadêmico na BD;</li> <li>✓ Existência do cadastro de matrículas na BD;</li> <li>✓ Regime do curso (semestral ou anual);</li> <li>✓ Regime de ingresso do acadêmico (vestibular ou transferência);</li> <li>✓ Existência de oferta de disciplinas para o semestre corrente;</li> <li>✓ Existência de turmas práticas e teóricas para as disciplinas ofertadas;</li> <li>✓ Quantidade de vagas por disciplina;</li> <li>✓ Frequência mínima (75%);</li> <li>✓ Notas de avaliação das disciplinas - 0 a 10;</li> <li>✓ Média mínima de aprovação 7,0;</li> <li>✓ Situação atual do acadêmico (calouro, aprovado sem dependência, aprovado com dependência (reprovado até 2 disciplinas), retido: reprovado em +2 disciplinas, retido: reprovado em dependência, trancado, formado, jubilado, transferido, reabertura de curso);</li> <li>✓ Situação atual do acadêmico em relação às disciplinas em curso e cursadas (aprovado ou reprovado);</li> <li>✓ Data da última matrícula;</li> <li>✓ Pré-requisitos das disciplinas;</li> </ul>

**Tabela 2 – Fatos da Sociedade de Agentes do SISMAT**

AÇÕES
<ul style="list-style-type: none"> <li>✓ Acessar dados cadastrais do acadêmico;</li> <li>✓ Acessar dados do histórico escolar do acadêmico (notas, frequência, data que cursou, etc.);</li> <li>✓ Acessar dados da matrícula;</li> <li>✓ Acessar dados da grade curricular (fluxograma) do curso;</li> <li>✓ Definir situação atual do acadêmico;</li> <li>✓ Definir pré-requisitos das disciplinas;</li> <li>✓ Mostrar dados no console;</li> <li>✓ Compor matrícula do acadêmico;</li> <li>✓ Atualizar BD;</li> </ul>

**Tabela 3 – Ações da Sociedade de Agentes do SISMAT**

DECISÕES
<ul style="list-style-type: none"> <li>✓ Se o acadêmico existe, transferir os dados cadastrais para o <i>site</i> do agente;</li> <li>✓ Se já cursou alguma disciplina, transferir histórico escolar para o <i>site</i> do agente;</li> <li>✓ Se matriculado atualmente, transferir dados da matrícula para o <i>site</i> do agente;</li> <li>✓ Transferir grade curricular (fluxograma) do curso para o <i>site</i> do agente;</li> <li>✓ Decidir qual a situação atual do acadêmico;</li> <li>✓ Decidir em Quais disciplinas o acadêmico pode matricular-se;</li> <li>✓ Se solicitado, compor a matrícula;</li> <li>✓ Se solicitado, atualizar BD;</li> <li>✓ Se findo o procedimento, encerrar a sessão;</li> </ul>

**Tabela 4 – Decisões da Sociedade de Agentes do SISMAT**

A seguir, faz-se uma descrição do comportamento básico de cada um dos agentes que compõem a sociedade de agentes do SISMAT.

#### 4.4 Agente Usuário (AU)

O Agente Usuário é o responsável por iniciar o processo do SISMAT, manifestando seu desejo de efetuar uma matrícula. No SISMAT, o AU é visto como um agente que participa ativamente do processo, pois através do Agente Interface, o usuário interage com a sociedade de agentes, buscando conhecimento sobre os acadêmicos, transmitindo suas intenções e solicitando a execução de tarefas à sociedade de agentes.

#### 4.5 Agente Interface (AI)

Este agente tem como principal objetivo permitir a comunicação do AU com os demais membros da sociedade. Seu comportamento reativo é permanecer em estado de espera, aguardando o AU expressar suas intenções.

Segundo Maes, agentes de interface são programas computacionais que empregam técnicas de inteligência artificial para prover assistência ao usuário, lidando com uma aplicação particular. A metáfora seria de um assistente pessoal que está colaborando com o usuário no mesmo ambiente de trabalho [MAES94].

O AU expressa sua intenção em efetuar uma matrícula através do RA do acadêmico. Este conhecimento é recebido pelo AI e representado por uma crença, porém, como ele não tem a capacidade de validá-lo, transmite este código ao AM, expressando o seu desejo de que a validação seja realizada.

Conforme a crença do AM (sobre o código), o AI pode tomar a decisão de solicitar um novo código ao AU (código inválido), caso contrário, enviar uma mensagem ao AM para que o mesmo mostre todo o conhecimento sobre o acadêmico.

Se o desejo do AU for matricular o acadêmico, ele transmite esta intenção ao AI através de uma resposta afirmativa na console, cujo fato é transmitido ao AM, que atuará de forma tal que a matrícula seja efetuada.

Para que uma matrícula seja realizada, o comportamento do AI estará relacionado a vários estados mentais. Na tabela 5, estão descritos os principais (crenças, compromissos e capacidades - conforme o paradigma AOP), considerados para modelagem deste agente, de modo a permitir uma melhor compreensão de como o agente é implementado utilizando estes conceitos. A seguir, são discutidos alguns detalhes de implementação do AI.



<b><u>AGENTE INTERFACE</u></b>		
<b>CRENÇAS</b>	<b>COMPROMISSOS</b>	<b>CAPACIDADES</b>
<ul style="list-style-type: none"> <li>✓ Sobre a existência do AU, AM e do AMT;</li> <li>✓ No conhecimento transmitido pelo AU;</li> <li>✓ AM pode validar os conhecimentos transmitidos pelo AU;</li> <li>✓ O acadêmico pode ser matriculado;</li> <li>✓ AM pode compor a matrícula;</li> <li>✓ AM pode mostrar o resultado da matrícula ao AM;</li> <li>✓ AM pode atualizar a BD;</li> <li>✓ AU pode solicitar a composição de uma matrícula.</li> <li>✓ AU pode não solicitar a composição de uma matrícula;</li> <li>✓ AU pode confirmar uma matrícula;</li> <li>✓ AU pode não confirmar uma matrícula;</li> </ul>	<ul style="list-style-type: none"> <li>✓ com o AU de buscar todos os conhecimentos possíveis a respeito do acadêmico;</li> <li>✓ com o AU de mostrar todos conhecimentos obtidos sobre o acadêmico;</li> <li>✓ com o AU de solicitar a matrícula do acadêmico, desde que o AU demonstre esta intenção;</li> <li>✓ com o AU de mostrar o resultado da matrícula pronta, ou as mensagens relativas a esta;</li> <li>✓ com o AU de atualizar as Bases de Dados desde que a matrícula tenha sido confirmada;</li> <li>✓ com o AM, sobre o encerramento de uma matrícula, avisá-lo para descartar os conhecimentos relativos ao acadêmico em questão;</li> </ul>	<ul style="list-style-type: none"> <li>✓ solicitar e receber o código do acadêmico do AU;</li> <li>✓ solicitar a composição da matrícula ao AM;</li> <li>✓ confirmar a intenção do AU em matricular o acadêmico;</li> <li>✓ confirmar a intenção do AU em não matricular o acadêmico;</li> <li>✓ solicitar ao AM para que providencie a atualização das BDs;</li> <li>✓ comunicar-se com o AU;</li> <li>✓ comunicar-se com o AM;</li> </ul>

**Tabela 5 – Estados Mentais Agente Interface**

A figura 3 apresenta o momento em que o AI verifica se o AM crê que o RA é válido. Sendo verdadeira esta crença, o AI solicita (*achieve*) ao AM que mostre o cadastro do acadêmico. Na figura 4, descreve-se o processo em que o AI já possui a crença de que o AU deseja efetuar a matrícula, então transmite (*tell*) esta crença ao AM para que o mesmo providencie a composição da matrícula.

Na descrição destas partes de programa das figuras 3 e 4, evidencia-se o uso de estados mentais nas decisões dos agentes, e observamos também que as suas decisões são tomadas não somente com base em seu estado mental, mas também de outro agente da sociedade, neste caso, o Agente Mediador.

```
// (R4). Se o RA é válido (crença do AM) então o AI solicita ao AM
// para que o mesmo mostre os conhecimentos oriundos da BD que estão
// em sua base de conhecimento.

IF
  BELIEF: AT ?now: cre_valida_ra(ra: ?ra);
THEN
  COMMITMENT_TO AU:
    BEGIN_AT ?now:
      achieve(sender:    #my-name,
              receiver:  Mediador,
              reply-with: mostrar_conhecimento,
              language:  LALO,
              ontology:   ra,
              content:    BEGIN_AT ?now: mostra_cadastro(ra: ?ra));
```

**Figura 3 – AI solicita tarefa ao AM**

```
// (R8). Quando o AU expressa seu desejo de matricular o acadêmico,
// ao AI este transmite esta crença ao AM.

IF
  BELIEF: AT ?now: cre_quer_matricular(ra: ?ra);
THEN
  COMMITMENT_TO Mediador:
    BEGIN_AT ?now:
      tell (sender:    #my-name,
            receiver:  Mediador,
            reply-with: quer_matricular,
            language:  LALO,
            ontology:   ra,
            content:    AT ?now: cre_quer_matricular(ra: ?ra));
```

**Figura 4 – AI transmite crença ao AM**

## 4.6 Agente Mediador (AM)

O principal objetivo deste agente é deixar transparente para o agente matriculador o acesso às informações necessárias ao processo de matrícula, desta forma ele é uma interface entre os sistemas de *software* (AMT e BD). Suas principais capacidades são: possuir o conhecimento de como e onde acessar as BDs, validar os conhecimentos extraídos destas e quando solicitado, transmitir este conhecimento ao AI e AMT.

Esta validação consiste em consultar as BDs, verificar se o acadêmico existe, a que curso pertence, acessar seu cadastro e histórico escolar. Posteriormente, com base em dados extraídos das BDs, definir a situação escolar do aluno.

Da mesma forma que o AI, o comportamento deste agente está diretamente relacionado às suas crenças, ou de acordo com aquelas transmitidas por outros agentes. Conforme a alteração destas durante um processo, suas atitudes podem mudar e determinadas tarefas podem ser realizadas ou não.

Quando iniciado, o seu comportamento básico é permanecer em estado de espera, aguardando alguma mensagem de outro agente. Normalmente, ele recebe a primeira mensagem do AI o qual informa sua crença sobre a intenção do AU em efetuar uma matrícula. Uma vez tendo esta crença em sua BC, toma a decisão de validar este conhecimento.

Se o AM, através das crenças do AI, passar a crer que o AU tem a intenção de que a matrícula seja realizada, toma a iniciativa de solicitá-la ao AMT, enviando uma mensagem ao mesmo.

Uma vez que o AM receba uma mensagem do AMT, sobre o sucesso da composição da matrícula, comunica este fato ao AI. A partir deste momento, caso o AI solicite a mostragem da matrícula, o AM envia uma mensagem ao AMT para que o mesmo exiba a matrícula na console.

Após uma visualização pelo AU do resultado da matrícula e caso este opte por confirmá-la, o AI envia outra mensagem ao AM para que o mesmo efetue uma atualização da BD, o qual por não ter esta capacidade, repassa este pedido ao AMT, para que o mesmo execute esta tarefa.

Do mesmo modo que ocorre com outros agentes da sociedade do SISMAT, o AM possui seus próprios estados mentais, os quais estão descritos na tabela 6, e para uma melhor exemplificação do uso do paradigma AOP neste agente, as figuras 5 e 6 descrevem partes de programa do AM. Na figura 5, observa-se o momento em que o AM verifica a crença transmitida pelo AI, a respeito da intenção do AU em efetuar a matrícula. Esta crença, se verdadeira, faz com que o AM cumpra dois compromissos (*commitment*) com o AI. O primeiro (figura 5) é definir o local das BDs, e inserir este conhecimento em sua BC; o segundo (figura 6) é transmitir ao AMT os seguintes conhecimentos: a crença sobre a intenção do AU em efetuar a matrícula, o RA do acadêmico e o local onde se encontram as BDs relativas a este aluno.

Neste exemplo do uso de AOP, percebe-se que o comportamento do AM foi influenciado por suas próprias crenças e pelas de outro agente, e a sua decisão foi cumprir compromissos assumidos com o AI.

```

// (R14). Quando o AI pedir para compor a matricula ao AM,
// primeiramente o AM determina o local onde as BDs estão.

IF
  BELIEF: AT ?now: cre_quer_matricular(ra: ?ra);
THEN
  COMMITMENT_TO Interface:
    BEGIN_AT ?now: insere_local(ra: ?ra);

// (R15). Neste ponto o local das BDs é inserido na BC do AM.

IF
  EXECUTING:
    insere_local(ra: ?ra);
THEN SUCCEED;

```

**Figura 5 – AM define local das BDs**

```

// (R16). Uma vez que o AI tenha solicitado a matricula, e o local das
// BDs tenha sido definido, o AM informa esta crença ao AMT, desta
// forma está solicitando ao matriculador a composição da matrícula.

IF
  BELIEF: AT ?now: cre_fazer_matricula(ra: ?ra, local: ?local);
THEN
  COMMITMENT_TO Interface:
    BEGIN_AT ?now:
      tell (sender:      #my-name,
           receiver:     Matriculador,
           in-reply-to:  fazer_matricula,
           language:     LALO,
           ontology:     matricula,
           content: AT ?now: cre_fazer_matricula(ra: ?ra, local: ?local));

```

**Figura 6 – AM solicita matrícula ao AMT**

<b>AGENTE MEDIADOR</b>		
<b>CRENÇAS</b>	<b>COMPROMISSOS</b>	<b>CAPACIDADES</b>
<ul style="list-style-type: none"> <li>✓ sobre a existência da BD;</li> <li>✓ sobre o local das BDs;</li> <li>✓ sobre a existência do AU, AI e AMT;</li> <li>✓ sobre onde buscar determinados conhecimentos na BD;</li> <li>✓ sobre o regime da instituição em relação aos seus critérios de matrícula;</li> <li>✓ sobre o código do acadêmico de como ele é composto e quais conhecimentos que podem ser extraídos dele;</li> </ul>	<ul style="list-style-type: none"> <li>✓ com o AI de validar o código do acadêmico;</li> <li>✓ com o AI de buscar conhecimento sobre o acadêmico na BD;</li> <li>✓ com o AI de definir a situação do acadêmico;</li> <li>✓ com o AI de mostrar o cadastro do acadêmico ao AU;</li> <li>✓ com o AI de mostrar o histórico escolar do acadêmico ao AU;</li> <li>✓ com o AI de mostrar a situação do acadêmico ao AU;</li> <li>✓ com o AI de informá-lo se o acadêmico pode ou não ser matriculado;</li> <li>✓ com o AI de descartar os conhecimentos relativos a um determinado acadêmico;</li> <li>✓ com o AI de solicitar a composição da matrícula ao AMT;</li> <li>✓ com o AMT de informá-lo do local das BDs;</li> <li>✓ com o AI de informá-lo se a matrícula foi realizada com sucesso ou não;</li> <li>✓ com o AI de solicitar ao AMT para que o mesmo mostre o resultado da matrícula ao AU;</li> <li>✓ com o AI para que as BDs sejam atualizadas caso o AU confirme a matrícula. (solicita ao AMT);</li> <li>✓ com o AMT de informá-lo que o AU não confirmou a matrícula;</li> </ul>	<ul style="list-style-type: none"> <li>✓ localizar as BDs;</li> <li>✓ localizar acadêmicos nas BDs;</li> <li>✓ localizar cadastro dos acadêmicos;</li> <li>✓ localizar histórico escolar dos acadêmicos;</li> <li>✓ localizar grades curriculares nas BDs;</li> <li>✓ localizar regimento da instituição;</li> <li>✓ buscar conhecimentos nas BDs;</li> <li>✓ validar RA;</li> <li>✓ extrair conhecimento do RA;</li> <li>✓ definir a situação do acadêmico;</li> <li>✓ descartar o conhecimento sobre um determinado acadêmico;</li> <li>✓ transmitir para o AMT o local das BDs;</li> <li>✓ mostrar o cadastro do acadêmico para o AU;</li> <li>✓ mostrar a situação do acadêmico ao AU;</li> <li>✓ mostrar o histórico escolar do acadêmico ao AU;</li> <li>✓ comunicar-se com o AI;</li> <li>✓ comunicar-se com o AMT;</li> </ul>

Tabela 6 – Estados Mentais Agente Mediador

## 4.7 Agente Matriculador (AMT)

Este é um agente essencialmente cognitivo capaz de, através de análises no histórico escolar do acadêmico, elaborar uma matrícula para o próximo período escolar, ou determinar se este acadêmico possui qualquer motivo que o impeça de ser matriculado.

Da mesma forma que o AM, o seu comportamento básico é permanecer em estado de espera, aguardando alguma mensagem de outro agente.

Em um processo normal de matrícula, o AMT recebe do AM uma mensagem com o seguinte conteúdo: a crença do AM em relação a intenção do AU em efetuar a matrícula; o RA do acadêmico e o local onde se encontram as informações relativas a este aluno.

Uma vez recebida esta mensagem, o procedimento deste agente é buscar todos os conhecimentos necessários sobre o acadêmico em questão, validar estes conhecimentos, definir a situação escolar do mesmo e, através de uma análise em todos estes conhecimentos, compor uma matrícula, ou justificar o impedimento deste em matricular-se.

O comportamento do AMT, assim como dos demais agentes da sociedade, também está relacionado a seus próprios estados mentais (descritos na tabela 7) e de outros agentes. Na figuras 7, 8 e 9 pode-se observar a implementação de alguns destes estados mentais, através de parte de programa do AMT. Três situações são mostradas, que envolvem alguns passos da composição de uma matrícula. Na primeira situação (figura 7), o AMT verifica sua crença sobre a intenção do AU em efetuar a matrícula. Se o AMT possuir esta crença, ele se compromete com o AM de validar o código do acadêmico. Para o AMT possuir esta crença, é necessário que em algum momento anterior, o AM tenha lhe comunicado este fato (mensagem KQML).

```
// (R1). Crença do AMT sobre a intenção do AU em matricular-se.
IF
  BELIEF: AT ?now: cre_fazer_matricula(ra: ?ra, local: ?local);
THEN
  COMMITMENT_TO Mediador:
    BEGIN_AT ?now: valida_ra(ra: ?ra, local: ?local);
```

Figura 7 – AMT valida ra

Na segunda situação (figura 8), o AMT de acordo com sua crença de que o aluno pode ser matriculado, cumpre o compromisso assumido com o AM de compor a matrícula.

```
// (R11). Se o AMT crê que o acadêmico pode ser matriculado
// então efetua a matrícula.

IF
  BELIEF: AT ?now: cre_pode_matricular(ra: ?ra);
THEN
  COMMITMENT_TO Mediador:
    BEGIN_AT ?now: compor_matricula(ra: ?ra);
```

**Figura 8 – AMT compõe a matrícula**

Finalmente, na figura 5, o AMT transmite ao AM sua crença sobre o sucesso do processo de matrícula.

```
// (R13). Uma vez composta a matrícula informa este fato ao AM.

IF
  BELIEF: AT ?now: cre_compos_matricula(ra: ?ra);
THEN
  COMMITMENT_TO Mediador:
    BEGIN_AT ?now:
      tell (sender:      #my-name,
            receiver:    Mediador,
            in-reply-to: fazer_matricula,
            language:    LALO,
            ontology:     matricula,
            comment:      "Matricula efetivada",
            content:      AT ?now: cre_compos_matricula(ra: ?ra));
```

**Figura 9 – AMT transmite crença ao AM**

Pode-se dizer que os procedimentos descritos anteriormente, apresentam as etapas do processo de matrícula, nas quais os resultados das aplicações dos estados mentais de todos os agentes do SISMAT atingem seu principal objetivo, “a elaboração da matrícula, satisfazendo o desejo do AU”.

Novamente, o uso do paradigma AOP destaca a sua importância no desenvolvimento do SISMAT. Neste exemplo (figuras 7, 8 e 9) mostra-se o uso dos estados mentais propostos

por Shoham, da seguinte forma: através de *crenças*, o AMT toma a *decisão* de cumprir seus *compromissos* (obrigações), que são realizados através de suas *capacidades* (tarefas), com a finalidade de alcançar seu objetivo que é a composição da matrícula.

Nas seções anteriores apresentaram-se os agentes e seus estados mentais, e observaram-se características especiais do paradigma AOP. A seguir, serão discutidos outros aspectos importantes, como a tarefa de descarte de conhecimento.



<b><u>AGENTE MATRICULADOR</u></b>		
<b>CRENÇAS</b>	<b>COMPROMISSOS</b>	<b>CAPACIDADES</b>
<ul style="list-style-type: none"> <li>✓ Sobre a existência da BD;</li> <li>✓ Sobre a existência do AU, AI e AM;</li> <li>✓ sobre onde buscar determinados conhecimentos na BD;</li> <li>✓ sobre o regime da instituição em relação aos seus critérios de matrícula;</li> <li>✓ sobre o código do acadêmico de como ele é composto e quais conhecimentos que podem ser extraídos dele;</li> </ul>	<ul style="list-style-type: none"> <li>✓ com o AM de compor a matrícula;</li> <li>✓ com o AM de atualizar a BD;</li> <li>✓ com o AM de validar novamente o código do acadêmico;</li> <li>✓ consigo mesmo de transmitir para sua BD os conhecimentos sobre um determinado acadêmico;</li> <li>✓ com o AM de informá-lo caso o código do acadêmico seja inválido;</li> <li>✓ consigo mesmo de descartar o conhecimento sobre um determinado acadêmico;</li> <li>✓ consigo mesmo de definir a situação do acadêmico;</li> <li>✓ com o AM de informá-lo caso o acadêmico não possa ser matriculado;</li> <li>✓ com o AM de informá-lo quando a matrícula estiver pronta;</li> <li>✓ com o AM de informá-lo quando não conseguiu compor a matrícula;</li> <li>✓ com o AM de transmitir a matrícula pronta ao AI;</li> <li>✓ com o AM de informá-lo se realmente a matrícula foi transmitida ao AI;</li> <li>✓ com o AM de informá-lo se realmente as BDs foram atualizadas;</li> </ul>	<ul style="list-style-type: none"> <li>✓ localizar acadêmicos nas BDs;</li> <li>✓ localizar cadastro dos acadêmicos;</li> <li>✓ localizar histórico escolar dos acadêmicos;</li> <li>✓ localizar grades curriculares nas BDs;</li> <li>✓ localizar regimento da instituição;</li> <li>✓ buscar conhecimentos nas BDs;</li> <li>✓ composição da matrícula;</li> <li>✓ validar RA;</li> <li>✓ extrair conhecimento do RA;</li> <li>✓ definir situação do acadêmico;</li> <li>✓ atualizar as BDs;</li> <li>✓ descartar o conhecimento sobre um determinado acadêmico;</li> <li>✓ mostrar a matrícula pronta para o AU;</li> <li>✓ comunicar-se com o AM;</li> </ul>

Tabela 7 – Estados Mentais Agente Matriculador

## 4.8 Discussões

A tarefa descarte de conhecimento (figura 2), implementada no SISMAT, possui uma característica especial que está relacionada ao desenvolvimento de sistemas com AOP. Cada agente criado através do LALO possui uma BC, a qual é utilizada para que o mesmo possa tomar decisões, baseadas em sua noção de tempo, i.e. tomar decisões em fatos do presente, passado e futuro.

Relacionando a BC do LALO com a memória dos seres humanos, que procuram esquecer de fatos que não mais têm importância para suas atividades, o LALO possui um comando *<FORGET>*, através do qual é possível excluir fatos (lembranças) da BC, que o agente não mais necessita, ou que podem gerar conflitos e redundâncias.

Na figura 10 está descrito um exemplo do uso deste comando: o AM naquele momento verifica se possui alguma crença sobre a invalidade do código de RA atual. Caso seja inválido, o AM transmitirá sua crença ao AI e descartará as crenças relativas ao valor e a validação do RA daquele momento para trás.

Desta forma, os agentes do SISMAT, podem descartar (esquecer) conhecimentos sobre um acadêmico que não é mais aquele que está sendo matriculado em um determinado momento, ou substituí-los por outros diferentes.

Estes procedimentos são necessários para não gerar conflitos e redundâncias, permitindo uma melhor otimização das tarefas e redução de processamento. Por exemplo, tomando-se por base o seguinte acontecimento: o AM crê que um determinado RA é válido e pode ser matriculado; esta crença é transmitida ao AU através do AI, que solicita a realização da matrícula. Contudo, após o AM ter validado o RA, e definido que o acadêmico poderia ser matriculado, este aluno entra na condição de jubilado<sup>12</sup>. Este novo fato sobre o acadêmico, contraria a crença do AM e a intenção do AU, e a matrícula não deve mais ser realizada.

Para que uma matrícula irregular não seja feita, antes de compô-la, o AMT valida o RA e define novamente se a matrícula pode ser efetuada, antes da atualização da BD. Seguindo a linha de raciocínio do exemplo anterior, o AMT pode possuir uma crença contrária a do AM sobre um RA, a qual será transmitida pelo AMT ao AM, que assumirá a nova crença e esquecerá a anterior. Conseqüentemente, o seu comportamento também mudará, ou seja, de-

---

<sup>12</sup> Instituto da Jubilação – será recusada nova matrícula ao acadêmico que não concluir o curso completo de graduação, no prazo máximo fixado para a integralização do respectivo currículo [UEPG99].

verá comunicar o AI para que o mesmo alerte o AU deste novo fato.

Diante destas exemplificações sobre o que ocorre em um sistema de matrículas, é possível perceber a complexidade que se apresenta e também a necessidade da presença de atitudes humanas na solução de determinados problemas.

Desta forma, os agentes do SISMAT aplicam um certo grau de comportamento similar ao humano, através de seus estados mentais, auxiliando o usuário na realização de matrículas.

Na próxima seção são abordados exemplos de matrículas e aspectos operacionais do SISMAT, em que é demonstrada a facilidade de operação do sistema.

```
// (R4). Se o AM passa a crer que o RA é invalido, informa esta crença
// ao AI e descarta seus conhecimentos e crenças sobre este acadêmico.

IF
  BELIEF: AT ?now: NOT cre_valida_ra(ra: ?ra);
THEN
  COMMITMENT_TO Interface:
    BEGIN_AT ?now:
      tell (sender:      #my-name,
            receiver:    Interface,
            in-reply-to: ra validar,
            language:    LALO,
            ontology:     ra,
            comment:      "Registro academico invalido",
            content:      AT ?now: NOT cre_valida_ra(ra: ?ra));
      FORGET: BEFORE ?now: cre_valida_ra(ra: ?ra);
      FORGET: FROM ?now: cre_valida_ra(ra: ?ra);
      FORGET: BEFORE ?now: cre_valor_ra(ra: ?ra);
      FORGET: FROM ?now: cre_valor_ra(ra: ?ra);
```

**Figura 10 – AM descarta conhecimentos**

## **5 Resultados: SISMAT**

### **5.1 Filosofia de Operação**

A operação do SISMAT é extremamente simples, o AU deve entrar apenas com o RA do acadêmico e duas respostas afirmativas, ou não, conforme a sua intenção: uma é para responder se realmente deseja efetuar a matrícula; a outra é para confirmar, se a matrícula elaborada pelo SISMAT deve ser gravada na BD do acadêmico. Findo este procedimento, o AI coloca-se novamente em estado de espera, aguardando outro RA.

Uma vez que o AU tenha expressado sua intenção de efetuar uma matrícula através do RA de um acadêmico, os agentes do SISMAT analisam este RA. Neste RA, estão expressos diversos conhecimentos em sua estrutura, (descritos na figura 11) que permitem identificar o acadêmico, e através desta identificação, buscar todos os conhecimentos sobre o mesmo (como: situação escolar, cadastro, histórico escolar, matrículas anteriores, etc.), e finalmente, determinar se pode ser ou não matriculado.

A seguir, são descritos exemplos de matrículas, que envolvem duas situações que os acadêmicos podem possuir: uma para o caso de um acadêmico sem dependência e outra com dependência. Estes exemplos são tratados sob a ótica de um usuário, nos quais são mostrados os conhecimentos que o SISMAT exibe ao AU, bem como o comportamento de seus agentes diante do usuário.

### **5.2 Exemplo de Matrícula: Acadêmico sem dependência**

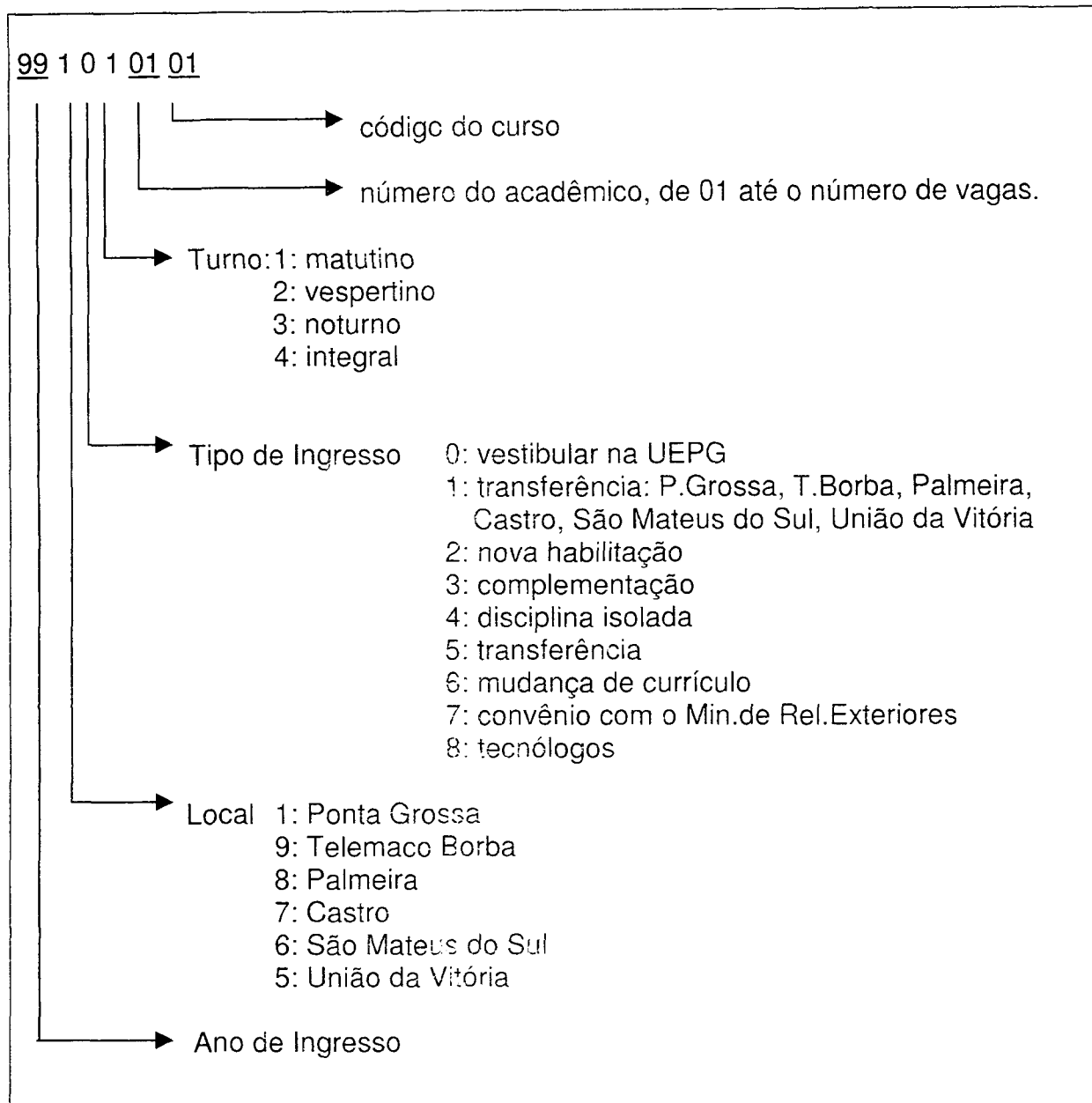
Nesta seção descreve-se o que ocorre no SISMAT, quando é transmitido para sua sociedade de agentes um RA, cujo acadêmico (hipotético) possui uma situação escolar regular, sem reprovações, conseqüentemente sem dependências e já tenha sido matriculado anteriormente.

A figura 12 mostra a interface exibida pelo AI, o qual coloca-se em estado de espera aguardando o AU expressar sua intenção de efetuar uma matrícula digitando um RA.

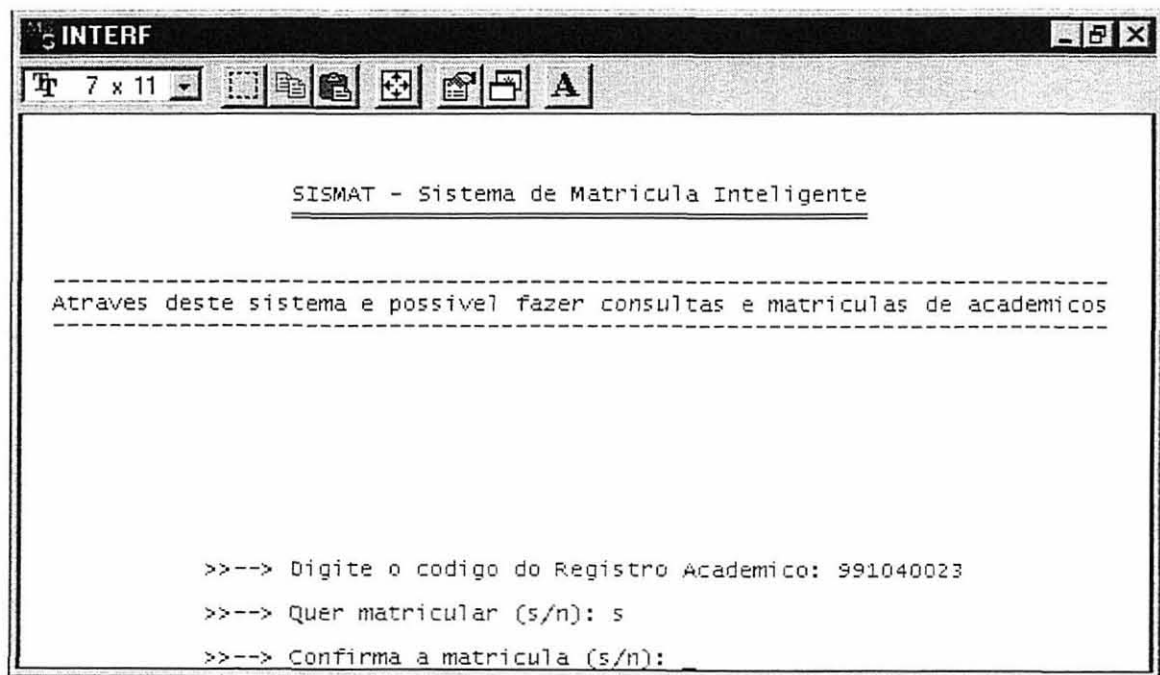
Nas figuras 13a e 13b, mostram-se os conhecimentos obtidos pelo AM, após o AU ter demonstrado sua intenção de efetuar uma matrícula. Nestas telas são exibidas informações cadastrais sobre o acadêmico, bem como a sua situação escolar e também o seu histórico es-

colar relativo a ultima matrícula. A partir deste momento, é solicitada ao AU uma confirmação sobre sua intenção de matricular o acadêmico em questão na próxima série (figura 12).

A figura 14 exibe a tela que corresponde ao resultado da composição de uma matrícula feita pelo AMT. Em seguida, na interface correspondente a figura 12 é solicitado ao AU para que o mesmo confirme a referida matrícula ou não de acordo com sua intenção. Caso a intenção do AU seja de confirmar a matrícula, as BDs serão atualizadas pelo AMT e esta matrícula passará a fazer parte do histórico escolar do acadêmico. Findo o procedimento de matrícula, o AI coloca-se novamente em estado de espera, aguardando o AU digitar um novo RA.



**Figura 11 – Estrutura Registro Acadêmico [UEPG99]**



**Figura 12 – Tela 1: Solicita RA.**

**MEDIAD**

SISMAT - Sistema de Matricula Inteligente

Cadastro do academico

Registro academico: 991040023  
 Nome: FERNANDA DA SILVA  
 Endereco: RUA DOS ESTUDOS, 111  
 Bairro: ESTUDANTIL  
 CEP: 84100  
 Cidade: PONTA GROSSA  
 Estado: PR  
 Telefone: 222-2222  
 Sexo: F  
 Data de nascimento: 11/11/80  
 Data de ultima matricula: 111998  
 Situacao atual: 5  
 Serie atual: 1  
 Turma pratica: 11  
 Turma teorica: 12

>>--> Dentro de alguns segundos seram exibidos: <--<<  
 >>--> a matricula anterior e a situacao atual <--<<

Figura 13a – Tela 2: Cadastro – Acadêmico s/dependência.

**MEDIAD**

Registro Academico: 991040023 - Nome: FERNANDA DA SILVA

Disciplinas cursadas na ultima matricula em 111998

Codigo -	Nome da disciplina	- conceito
501070	- Psicologia das Relacoes Humanas	A
203042	- Logica Computacional	A
101051	- Calculo Diferencial e Integral	A
203056	- Introd.a Investigacao Cientifica	A
203016	- Organizacao de Computadores	A
505027	- Ingles Tecnico	A
506044	- Portugues Instrumental	A
203015	- Estrutura de Programacao	A

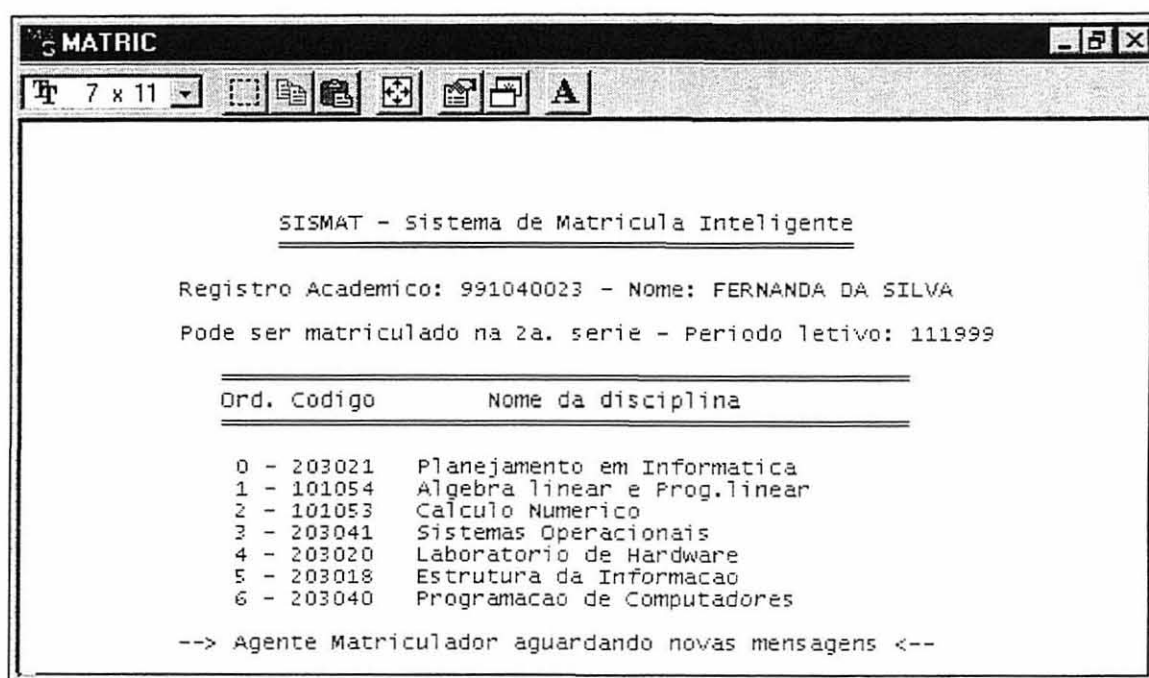
Situacao atual

Reprovadas por nota: 0 - Reprovadas por falta: 0 - Dependencias: 0  
 Situacao atual 5 - Pode ser matriculado na 2a.serie - Series do curso: 4

--> Agente Mediador aguardando novas mensagens <--

Figura 13b – Tela 2: Histórico e Situação Escolar– Acadêmico s/dependência.





**Figura 14 – Tela 3: Matrícula - Acadêmico s/dependência.**

### 5.3 Exemplo de Matrícula: Acadêmico com dependência

Entende-se por regime de dependência a faculdade de poder o aluno freqüentar até duas disciplinas em que ficou reprovado simultaneamente com a série imediatamente seguinte, para a qual será regularmente promovido [UEPG99].

O acadêmico que freqüentar alguma série do curso com disciplina(s) em dependência somente será promovido para série seguinte quando, além de atender às demais exigências, lograr aprovação nessa(s) disciplina(s) [UEPG99].

Tomando por base o exposto anteriormente sobre um acadêmico em regime de dependência, descreve-se nesta seção um exemplo de matrícula que contempla esta situação. Como o objetivo destes exemplos é criar a sensação que o usuário tem diante do SISMAT, utilizou-se a grade curricular do curso de Informática da UEPG, descrita na figura 15. Supondo-se que um acadêmico (hipotético), tivesse cursado a primeira série deste curso e no final do período letivo, em seu histórico escolar constam todas as disciplinas da primeira série, porém, nas disciplinas com códigos 501070 e 101051 (em destaque na figura 15) o mesmo não tenha logrado aprovação.

Neste caso, quando este acadêmico for matriculado na segunda série, ele será matriculado em todas as disciplinas da série correspondente mais as disciplinas da primeira série, nas quais não foi aprovado.

Descreve-se a seguir o comportamento dos agentes do SISMAT, para o caso anteriormente exemplificado, onde são destacadas as principais ocorrências.

Inicialmente, conforme já visto, o AU transmite o RA do acadêmico ao agentes do SISMAT, digitando este código em uma console (figura 12). Através deste RA, o acadêmico será identificado pelo AM, que fará uma análise em seu histórico escolar. Nesta análise, a situação de dependência será detectada e informada ao AU através da tela correspondente à figura 16. E finalmente através da tela correspondente à figura 17, o agente usuário tomará a decisão de matricular ou não o acadêmico conforme o seu desejo.

Ao fazer-se um relacionamento entre os dois exemplos (acadêmico sem/com dependência) anteriormente descritos, percebe-se que, independentemente da situação que os acadêmicos possam apresentar, o resultado ao usuário sempre será o mesmo, ou seja, o objetivo do sistema sempre é alcançado, que é realizar e/ou verificar se o acadêmico pode ser matriculado, sem a interferência do usuário.

Destaca-se nesta abordagem, que a sociedade de agentes do SISMAT cumpre o papel de indivíduos envolvidos em um processo de matrícula, que cooperam entre si aplicando seu conhecimento para que as suas metas sejam alcançadas.

São descritos pelas interface, os resultados que um usuário obtém do SISMAT, na realização de uma matrícula. Abordam-se a seguir quais são os estados mentais de acordo com o paradigma AOP, que ocorrem e são compartilhados entre os agentes desta sociedade para realização da matrícula.

BACHARELADO EM INFORMÁTICA									
1ª SÉRIE	Psicologia das Relações Humanas	Lógica Computacional	Cálculo Diferencial e Integral	Introdução à Investigação Científica	Organização de Computadores	Signos, Textos	Português Instrumental	Estrutura de Programação	
782 21	601070 6A 2	203042 102 3	101051 102 3	203056 6A 2	203016 136 4	505027 102 3	501044 6A 2	203015 136 4	
2ª SÉRIE	Planejamento em Informática	Álgebra Linear e Programação Linear	Cálculo Numérico	Sistemas Operacionais	Laboratório de Hardware	Estrutura de Informação	Programação de Computadores		
884 26	203021 102 3	101054 102 3	101052 102 3	203041 136 4	203020 102 3	203018 136 4	203040 204 5		
3ª SÉRIE	Análise de Sistemas	Gerenciamento de Centro de Processamento de Dados	Estatística	Legislação Aplicada à Informática	Banco de Dados	Inteligência Artificial	Geometria Analítica		
884 26	203025 170 5	203024 102 3	101055 136 4	603017 6A 2	203023 204 5	203026 102 3	101050 102 3		
4ª SÉRIE	Projeto	Estágio Supervisionado	Computação Gráfica	Teleprocessamento	Análise de Algoritmos				
680 20	203029 136 4	203032 272 8	203030 102 3	203027 102 3	203043 6A 2				

Em vigor a partir de 02 de janeiro de 1995 (Res. C.A. nº 265/94 e alterado pela Res. CEPE nº 54/96)

**Figura 15 – Grade curricular Curso de Informática [UEPG99].**

MS MEDIAD

7 x 11

Registro Academico: 991040023 - Nome: FERNANDA DA SILVA

Disciplinas cursadas na ultima matricula em 111998

Codigo -	Nome da disciplina	- conceito
501070 -	Psicologia das Relacoes Humanas	N
203042 -	Logica Computacional	A
101051 -	Calculo Diferencial e Integral	N
203056 -	Introd.a Investigacao Cientifica	A
203016 -	Organizacao de Computadores	A
505027 -	Ingles Tecnico	A
506044 -	Portugues Instrumental	A
203015 -	Estrutura de Programacao	A

Situacao atual

Reprovadas por nota: 2 - Reprovadas por falta: 0 - Dependencias: 0  
situacao atual 4 - Pode ser matriculado na 2a.serie - Series do curso: 4

--> Agente Mediador aguardando novas mensagens <--

Figura 16 – Tela 2: Cadastro e Histórico Escolar – Acadêmico c/dependência.

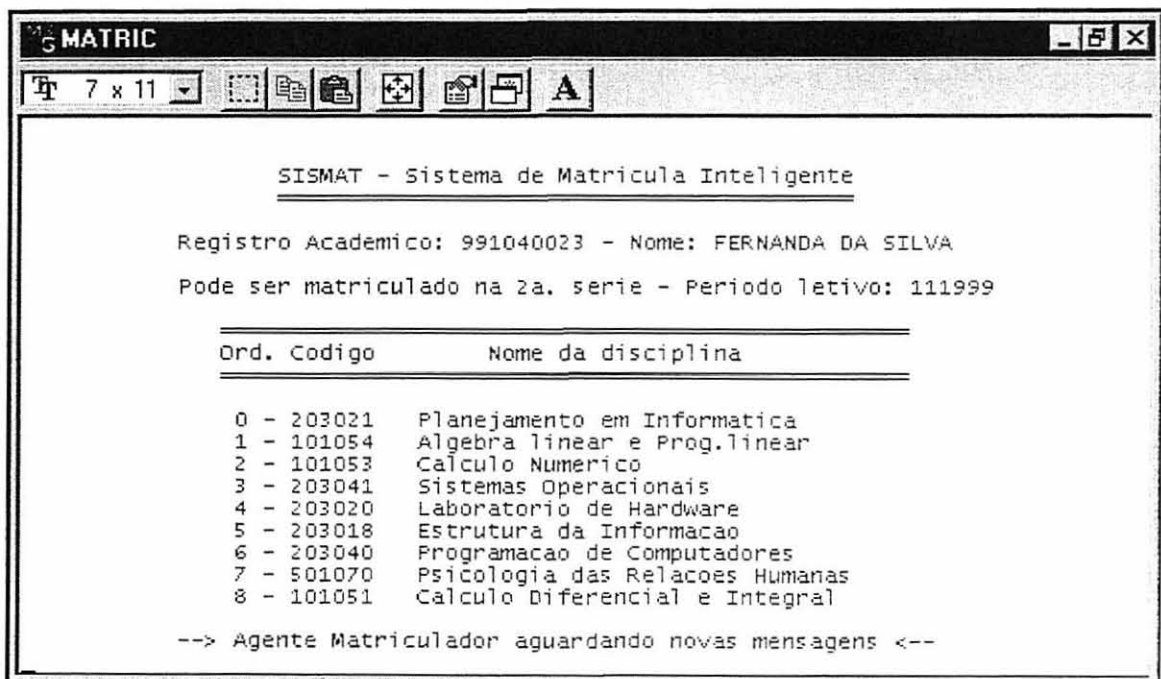


Figura 17 – Tela 3: Matrícula – Acadêmico c/dependência.

## **5.4 Estados mentais envolvidos em uma matrícula**

Far-se-á nesta seção uma amostragem dos estados mentais que ocorrem em cada um dos agentes do SISMAT, quando da composição de uma matrícula. O exemplo a ser seguido trata-se de um acadêmico com uma situação escolar regular, sem reprovações e sem dependências. São exibidos os estados mentais sob a forma da implementação efetuada em LALO com as respectivas descrições de cada ocorrência.

### **5.4.1 Estados Mentais AI**

Na figura 18 são observadas as crenças que o AI pode possuir durante este processo. Como o AI tem por finalidade principal estabelecer a comunicação do AU com a sociedade de agentes do SISMAT (figura 1), as suas interações e comportamentos serão afetados pelo AU e AM. Portanto suas crenças serão baseadas nas intenções do AU e nas crenças que o AM possuirá com relação ao acadêmico a ser matriculado. Por tratar-se de um processo normal de matrícula, em que: o AU tem a intenção de que realmente o acadêmico seja matriculado, o RA é válido, o acadêmico possui um histórico escolar regular e pode ser matriculado. Portanto, todas as crenças neste caso serão verdadeiras.

Na figura 19 observam-se quais compromissos que o AI deverá cumprir para que o objetivo da sociedade de agentes seja alcançado. Da mesma forma como ocorre nas crenças, os compromissos assumidos são com relação ao AU e AM.

Neste processo de matrícula e de acordo com a modelagem do AI, a figura 20 relaciona as tarefas (capacidades) que o AI deve realizar para que a matrícula seja efetuada. Estas tarefas basicamente referem-se ao processo de comunicação (interface) que o AI estabelece com o AU.

AGENTE INTERFACE	
<u>CRENÇAS</u>	<u>DESCRIÇÃO</u>
AT ?now: cre_valor_ra(ra: ?ra)	Crença transmitida ao AM, sobre o RA informado pelo AU;
AT ?now: cre_valida_ra(ra: ?ra)	Crença transmitida pelo AM sobre a validação do RA;
AT ?now: cre_pode_matricular(ra: ?ra)	Crença transmitida pelo AM sobre a situação escolar do acadêmico que permite a matrícula;
AT ?now: cre_quer_matricular(ra: ?ra)	Crença transmitida ao AM, sobre a intenção do AU em matricular o acadêmico;
AT ?now: cre_compos_matricula(ra: ?ra)	Crença transmitida pelo AM, sobre o fato de que a matrícula já foi montada pelo AMT;
AT ?now: cre_AU_quer_ver_matricula(ra: ?ra)	Crença transmitida ao AM, sobre a intenção do AU em ver a matrícula;
AT ?now: cre_confirma_matricula(ra: ?ra)	Crença transmitida ao AM, sobre a intenção do AU em confirmar a matrícula e atualizar as BDs;
AT ?now: cre_atualiza_BD(ra: ?ra)	Crença transmitida pelo AM, confirmando a atualização das BDs.

Figura 18 – Crenças AI

AGENTE INTERFACE	
<u>COMPROMISSOS</u>	<u>DESCRIÇÃO</u>
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: tell(sender: #my-name, receiver: Mediador, reply-with: ra_validar, language: LALO, ontology: ra, content: AT ?now: cre_valor_ra(ra: ?ra));	Informar o AM sobre o RA transmitido pelo AU.
<b>COMMITMENT_TO AU:</b> BEGIN_AT ?now: achieve(sender: #my-name, receiver: Mediador, reply-with: mostrar_conhecimento, language: LALO, ontology: ra, content: BEGIN_AT ?now: mostra_cadastro(ra: ?ra));	Solicitar ao AM para que mostre o cadastro do acadêmico.
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: tell (sender: #my-name, receiver: Mediador, reply-with: quer_matricular, language: LALO, ontology: ra, content: AT ?now: cre_quer_matricular(ra: ?ra));	Transmitir ao AM, a intenção do AU em matricular o acadêmico;
<b>COMMITMENT_TO AU:</b> BEGIN_AT ?now: tell(sender: #my-name, receiver: Mediador, reply-with: quer_ver_matricula, language: LALO, ontology: ra, content: AT ?now: cre_AU_quer_ver_matricula(ra: ?ra));	Transmitir ao AM a intenção do AU de ver a matrícula pronta ao AM;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: tell (sender: #my-name, receiver: Mediador, reply-with: confirma_matricula, language: LALO, ontology: ra, content: AT ?now: cre_confirma_matricula(ra: ?ra));	Transmitir ao AM a intenção do AU em confirmar a matrícula;

Figura 19 – Compromissos AI



AGENTE INTERFACE	
<u>TAREFAS</u>	<u>DESCRIÇÃO</u>
PRIVATE ATOMIC entra_ra(ra);	Solicitar um RA ao AU;
PRIVATE ATOMIC solicita_matricula(ra);	Solicitar do AU a sua intenção em efetuar a matrícula;
PRIVATE ATOMIC confirma_matricula(ra);	Solicitar do AU a sua intenção em confirmar a matrícula e atualizar as BDs;

**Figura 20 – Tarefas AI**

#### 5.4.2 Estados Mentais AM

Do mesmo modo como foram descritos os estados mentais do AI. Na figura 21 pode-se observar quais são as crenças que o AM pode possuir durante a realização de uma matrícula para um acadêmico regular. No SISMAT, o AM atua como um gerente da sociedade, e suas interações são com o AI e o AMT (figura 1). Portanto, de acordo com as mensagens recebidas destes agentes, o AM formará as suas crenças, as quais poderão ser transmitidas aos remetentes ou a outro agente conforme o caso, por exemplo: uma mensagem oriunda do AI informando sua crença sobre a intenção do AU em efetuar uma matrícula, é recebida pelo AM e transmitida ao AMT para que este componha a matrícula.

Na figuras 22a e 22b pode-se observar quais os compromissos que devem ser cumpridos pelo AM para o exemplo corrente, e na figura 23 quais são as tarefas que devem ser realizadas.

Percebe-se que a totalidade dos compromissos do AM são com o AI, o motivo disto, é que a sociedade busca resolver problemas de um usuário, e as interações do usuário com o SISMAT fazem-se através do AI, que é o intermediário neste processo.

Como observado na figura 23, as tarefas que o AM deve executar, expressam principalmente a sua capacidade de mediação no SISMAT, pois ele valida conhecimentos, determina locais onde encontram-se estes conhecimentos, busca-os, mostra-os, e de uma forma geral, o AM evita que sejam feitas tentativas de matricular acadêmicos quando estes não podem ser

matriculados, ou ainda quando simplesmente o objetivo do AU seja apenas uma consulta, ou uma previsão de matrícula.

AGENTE MEDIADOR	
<u>CRENÇAS</u>	<u>DESCRIÇÃO</u>
AT ?now: cre_valor_ra(ra: ?ra);	Crença transmitida ao AI, sobre o RA informado pelo AU;
AT ?now: cre_valida_ra(ra: ?ra);	Crença sobre o resultado da validação do RA;
AT ?now: cre_pode_matricular(ra: ?ra);	Crença resultante da análise do histórico escolar do acadêmico, sobre a sua situação escolar;
AT ?now: cre_quer_matricular(ra: ?ra);	Crença transmitida pelo AI, sobre a intenção do AU em matricular o acadêmico;
AT ?now: cre_fazer_matricula (ra: ?ra, local: ?local);	Crença sobre a intenção do AU em fazer a matrícula e sobre o local onde encontram-se os dados do acadêmico;
AT ?now: cre_compos_matricula(ra: ?ra);	Crença transmitida pelo AMT, informando que a matrícula já foi montada;
AT ?now: cre_AU_quer_ver_matricula(ra: ?ra);	Crença transmitida pelo AI, sobre a intenção do AU em ver a matrícula;
AT ?now: cre_mostra_matricula(ra: ?ra);	Crença transmitida pelo AMT, confirmando que a matrícula já foi mostrada ao AU.
AT ?now: cre_confirma_matricula(ra: ?ra);	Crença transmitida pelo AI, informando que o AU confirma a matrícula e quer atualizar as BDs;
AT ?now: cre_atualiza_BD(ra: ?ra);	Crença transmitida pelo AMT, informando que as BDs foram atualizadas com sucesso;

**Figura 21 -- Crenças AM**

AGENTE MEDIADOR	
<u>COMPROMISSOS</u>	<u>DESCRIÇÃO</u>
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: valida_ra(ra: ?ra);	Validar o RA.
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: busca_conhecimento_BD(ra: ?ra); BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Interface, in-reply-to: ra_validar, language:    LALO, ontology:    ra, comment:     "Registro academico valido", content:     AT ?now: cre_valida_ra(ra: ?ra));	Buscar os conhecimentos na BD, e transmitir o resultado da validação;
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: define_situacao(ra: ?ra); BEGIN_AT ?now: pode_matricular(ra: ?ra);	Definir a situação escolar e verificar se o acadêmico pode ser matriculado;
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: mostra_cadastro(ra: ?ra);	Mostrar as informações cadastrais do acadêmico;
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: mostra_historico(ra: ?ra); BEGIN_AT ?now: mostra_situacao(ra: ?ra);	Mostrar o histórico escolar e a situação do acadêmico;
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Interface, in-reply-to: pode_matricular, language:    LALO, ontology:    matricula, content: AT ?now: cre_pode_matricular(ra: ?ra));	Informar o AI sobre a situação escolar do acadêmico;
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Matriculador, in-reply-to: fazer_matricula, language:    LALO, ontology:    matricula, content:     AT ?now: cre_fazer_matricula(ra: ?ra, local: ?local));	Transmitir ao AMT, a intenção do AU em fazer a matrícula, o RA e o local das BDs relativas ao acadêmico;
<b>COMMITMENT_TO Interface:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Interface, in-reply-to: fazer_matricula, language:    LALO, ontology:    matricula, comment:     "Matricula efetivada", content:     AT ?now: cre_compos_matricula(ra: ?ra));	Informar ao AI que a matrícula já foi montada;

Figura 22a – Compromissos AM

AGENTE MEDIADOR	
<u>COMPROMISSOS</u>	<u>DESCRIÇÃO</u>
<b>COMMITMENT TO Interface:</b> BEGIN_AT ?now: achieve(sender:       #my-name, receiver:   Matriculador, reply-with: mostre_matricula, language:   LALO, ontology:   matricula, content: BEGIN_AT ?now: mostra_matricula(ra: ?ra));	Solicitar ao AMT que mostre a matrícula montada;
<b>COMMITMENT TO Interface:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Interface, in-reply-to: mostra_matricula, language:    LALO, ontology:    matricula, comment:     "Amostragem efetivada", content: AT ?now: cre_mostra_matricula(ra: ?ra));	Informar ao AI que a matrícula já foi mostrada ao AU;
<b>COMMITMENT TO Interface:</b> BEGIN_AT ?now: achieve(sender:       #my-name, receiver:   Matriculador, reply-with: mostre_matricula, language:   LALO, ontology:   matricula, content: BEGIN_AT ?now: atualiza_BD(ra: ?ra));	Solicitar ao AMT a atualização das BDs;
<b>COMMITMENT TO Interface:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Interface, in-reply-to: atualizacao_BD, language:    LALO, ontology:    matricula, comment:     "Atualizacao efetivada", content: AT ?now: cre_atualiza_BD(ra: ?ra));	Informar o AI que as BDs já foram atualizadas com sucesso;

Figura 22b – Compromissos AM

AGENTE MEDIADOR	
<u>TAREFAS</u>	<u>DESCRIÇÃO</u>
PRIVATE ATOMIC valida_ra(ra);	Validar um RA;
PRIVATE ATOMIC busca_conhecimento_BD(ra);	Buscar conhecimentos nas BDs;
PRIVATE ATOMIC busca_conhecimento_BD(ra);	Buscar conhecimentos nas BDs;
PRIVATE ATOMIC define_situacao(ra);	Definir a situação escolar do acadêmico
PRIVATE ATOMIC pode_matricular(ra);	Verificar se o acadêmico pode ser matriculado;
PUBLIC ATOMIC mostra_cadastro(ra);	Mostrar na tela o cadastro do acadêmico;
PUBLIC ATOMIC mostra_historico(ra);	Mostrar na tela o histórico escolar do acadêmico;
PUBLIC ATOMIC mostra_situacao(ra);	Mostrar na tela a situação escolar do acadêmico;
PRIVATE ATOMIC insere_local(ra);	Definir e colocar em sua BC o local onde se encontram todos os dados do acadêmico e aqueles necessários ao processo de matrícula;
PRIVATE ATOMIC deve_esquecer(ra);	Eliminar de sua BC, todos os conhecimentos sobre um determinado processo de matrícula;

**Figura 23 – Tarefas AM**

### 5.4.3 Estados Mentais AMT

O AMT por ser o agente que efetua a matrícula, recebe mensagens somente do AM (figura 1), as quais trazem em seu conteúdo, crenças relativas às intenções do AU e aquelas formadas pelo próprio AM a respeito do acadêmico, descritas na figura 24. Porém o AMT, antes de compor a matrícula, verifica se as crenças do AM em relação ao acadêmico são verdadeiras. Como já visto no capítulo 5, isto acontece em função de alguma alteração que possa ocorrer nas BDs, por este motivo a situação escolar do acadêmico pode ser alterada, contrariando alguma crença do AM.

Da mesma forma como ocorre em relação às crenças, os compromissos do AMT (descritos na figura 25), são com o AM, apesar de que, indiretamente estes compromissos atendem as intenções do AU, ou ainda pode-se considerar compromissos que atendem as expectativas da sociedade como um todo.

Na figura 26, descrevem-se as tarefas que são necessárias para a composição de uma matrícula, independentemente da situação escolar do acadêmico, exceto para o caso de um acadêmico que não possa ser matriculado.

Nesta análise do comportamento de agentes modelados com estados mentais, percebe-se que, através do compartilhamento dos conhecimentos e das habilidades individuais de cada agente, a matrícula é efetuada. Novamente, surge a semelhança que os agentes possuem com atitudes humanas, através do uso de estados mentais, visto que, os modelos computacionais procuram simular de forma restrita o comportamento humano, utilizando uma metáfora dos estados mentais humanos.

Nas exemplificações desta seção, nota-se também que o domínio escolhido para implementação deste trabalho, é excelente para o desenvolvimento de SMA, e mais precisamente de agentes modelados com estados mentais.

A seguir, discute-se quais propriedades são encontradas nos agentes que compõem a sociedade do SISMAT, cujo desenvolvimento foi possível através da abordagem AOP.

AGENTE MATRICULADOR	
<u>CRENÇAS</u>	<u>DESCRIÇÃO</u>
AT ?now: cre_fazer_matricula(ra:?ra,local:?local);	Crença transmitida pelo AM, sobre a intenção do AU em matricular o acadêmico;
AT ?now: cre_valida_ra(ra: ?ra);	Crença sobre a sua validação do RA;
AT ?now: cre_pode_matricular(ra: ?ra);	Crença resultante da análise do histórico escolar do acadêmico, sobre a sua situação escolar;
AT ?now: cre_compos_matricula(ra: ?ra);	Crença transmitida ao AM, informando que a matrícula foi montada;
AT ?now: cre_mostra_matricula(ra: ?ra);	Crença transmitida ao AM, informando que a matrícula já foi mostrada para o AU;
AT ?now: cre_confirma_matricula(ra: ?ra);	Crença transmitida ao AM, sobre a intenção do AU em confirmar a matrícula e atualizar as BDs;
AT ?now: cre_atualiza_BD(ra: ?ra);	Crença transmitida ao AM, informando que as BDs foram atualizadas com sucesso;

Figura 24 – Crenças AMT

AGENTE MATRICULADOR	
<u>COMPROMISSOS</u>	<u>DESCRIÇÃO</u>
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: valida_ra(ra: ?ra, local: ?local);	Validar novamente o RA p/que não sejam realizadas matrículas irregulares;
<b>COMMITMENT_TO #myself:</b> BEGIN_AT ?now: busca_conhecimento_BD(ra: ?ra);	Buscar novamente as informações das BDs p/evitar matrículas irregulares;
<b>COMMITMENT_TO #myself:</b> BEGIN_AT ?now: define_situacao(ra: ?ra); BEGIN_AT ?now: pode_matricular(ra: ?ra);	Definir novamente a situação e verificar novamente se o acadêmico pode ser matriculado para evitar matrículas irregulares;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: compor_matricula(ra: ?ra);	Compor a matrícula;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Mediador, in-reply-to: fazer_matricula, language:    LALO, ontology:    matricula, comment:     "Matricula efetivada", content: AT ?now: cre_compos_matricula(ra: ?ra));	Informar ao AM que a matrícula já foi montada;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: mostra_matricula(ra: ?ra);	Mostrar a matrícula ao AU;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Mediador, in-reply-to: mostra_matricula, language:    LALO, ontology:    matricula, comment:     "Amostragem efetivada", content: AT ?now: cre_mostra_matricula(ra: ?ra));	Informar ao AM, que a matrícula já foi mostrada ao AU;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: atualiza_BD(ra: ?ra);	Atualizar as BDs;
<b>COMMITMENT_TO Mediador:</b> BEGIN_AT ?now: tell (sender:       #my-name, receiver:    Mediador, in-reply-to: atualizacao_BD, language:    LALO, ontology:    matricula, comment:     "Atualizacao efetivada", content: AT ?now: cre_atualiza_BD(ra: ?ra));	Informar ao AM, que as BDs foram atualizadas com sucesso;

Figura 25 – Compromissos AMT



AGENTE MATRICULADOR	
<u>TAREFAS</u>	<u>DESCRIÇÃO</u>
PRIVATE ATOMIC valida_ra(ra, local);	Validar um RA;
PRIVATE ATOMIC busca_conhecimento_BD(ra);	Buscar conhecimentos nas BDs;
PRIVATE ATOMIC busca_conhecimento_BD(ra);	Buscar conhecimentos nas BDs;
PRIVATE ATOMIC define_situacao(ra);	Definir a situação escolar do acadêmico
PRIVATE ATOMIC pode_matricular(ra);	Verificar se o acadêmico pode ser matriculado;
PUBLIC ATOMIC compor_matricula(ra);	Monta a matrícula de um acadêmico;
PUBLIC ATOMIC mostra_matricula(ra);	Mostrar na tela a matrícula pronta;
PUBLIC ATOMIC atualiza_BD(ra);	Atualizar as BDs;
PRIVATE ATOMIC deve_esquecer(ra);	Eliminar de sua BC, todos os conhecimentos sobre um determinado processo de matrícula;

**Figura 26 – Tarefas AMT**

## 5.5 Propriedades dos agentes do SISMAT

Pode-se afirmar que, uma das principais razões para adotar agentes inteligentes no desenvolvimento de sistemas baseados em conhecimento, reside nas propriedades que os mesmos podem possuir.

Através destas propriedades, os agentes podem exibir habilidades sociais, inerentes a um SMA, as quais definem o seu comportamento. Portanto é possível afirmar também, que o comportamento de um sistema é influenciado pelas propriedades encontradas nos agentes. Desta forma, para que o SISMAT pudesse funcionar adequadamente de acordo com a abordagem SMA, aos seus agentes foram atribuídas as seguintes propriedades:

- *Autonomia* - os agentes AM e o AMT tem um grau de autonomia que lhes permite realizar suas tarefas sem muita intervenção humana, pois eles possuem capacidades específicas e delimitadas que permitem decidir e realizar ações a fim de atingir seu objetivo, e podem reagir diferentemente a mesmas mensagens de acordo com seu estado mental. Por exemplo, se o AM solicita ao AMT que efetue a matrícula de um acadêmico, e o AMT crê que este aluno não pode ser matriculado, então ele não efetua esta matrícula com base em seu estado mental atual;
- *Comunicabilidade* - os agentes do SISMAT comunicam-se entre si via KQML;
- *Cooperação* - para que seja alcançado o objetivo do sistema, que é a efetivação da matrícula. Nenhum agente consegue individualmente efetuar a matrícula, para isto eles cooperam um com o outro para atingirem este objetivo;
- *Iniciativa* - os agentes do SISMAT não reagem somente em relação ao seu ambiente e estado mental, mas também em relação aos estados mentais de outros agentes, tomando a iniciativa em certos casos. Como exemplo, quando o AM tem conhecimento de que o AI possui a crença sobre a intenção do AU de efetuar uma matrícula, o AM toma a iniciativa de solicitar a composição desta ao AMT;
- *Reatividade* - os agentes reagem respondendo aos estímulos oriundos do ambiente ou de outros agentes, quando estes passam a ter conhecimento das crenças de outros agentes da sociedade, por exemplo o AM reage às crenças do AI sobre as intenções do AU em efetuar uma matrícula;

- *Sociabilidade* - o aspecto da cooperação já é um fato que justifica a propriedade da sociabilidade, bem como a troca de conhecimento que os agentes do SISMAT fazem, interagindo entre si para que o objetivo seja alcançado;
- *Reusabilidade* - os agentes do SISMAT podem ser reusáveis, pois podem servir a qualquer curso, embora os conhecimentos sejam diferentes em cada caso. Para que os agentes se adaptem a cada curso, os conhecimentos são extraídos das BD;

A seguir discute-se, os resultados obtidos com o desenvolvimento do LALO, bem como as abordagens utilizadas em sua implementação.

## 5.6 Discussões

A Inteligência Artificial tem como principal objetivo, a representação do comportamento humano através de modelos computacionais. Esta idéia é reforçada quando um sistema baseado em conhecimento é desenvolvido utilizando agentes inteligentes, modelados com estados mentais.

Neste contexto, o comportamento dos agentes do SISMAT está diretamente relacionado às suas crenças, ou de acordo com as crenças transmitidas por outros agentes. Conforme a alteração destas durante um processo, suas atitudes podem mudar e determinadas tarefas podem ser realizadas ou não. Por exemplo, quando o AM transmite ao AMT a intenção do AU em efetuar uma matrícula: ele (AM) possui a crença de que o acadêmico pode ser matriculado; contudo, antes de compor a matrícula, o AMT busca a sua própria crença, para saber se realmente a matrícula pode ser realizada. Esta situação pode ocorrer, devido ao intervalo de tempo entre o estabelecimento da crença do AM e o pedido efetivo de matrícula do AU. Neste espaço de tempo uma alteração na BD pode ocorrer (por exemplo, a troca da nota de uma disciplina que provoque a reprovação), motivando o impedimento da matrícula, o que contraria a crença do AM, porém, o AMT agirá de acordo com o que ele crê naquele momento, e não efetuará a matrícula.

Ao se desenvolver o SISMAT, percebeu-se que o paradigma AOP possui vantagens significativas sobre o OOP. Entre estas, destaca-se a noção de tempo que é atribuída aos agentes. Este aspecto permite uma maior flexibilização na modelagem e desenvolvimento de um sistema. O fator tempo associado ao agente, cria uma semelhança maior com atitudes humanas, pois permite o agente possuir uma memória [GIR99], isto é, pode-se introduzir a no-

ção de lembrança. O agente pode lembrar de fatos (crenças e decisões) ocorridos no passado, para sua tomada de decisão atual, bem como, lembrar de compromissos que foram assumidos anteriormente para serem executados no futuro.

O protótipo do SISMAT foi desenvolvido em LALO, o qual possui como características positivas: a sua arquitetura que é extensível, pois o desenvolvedor pode ampliar o conjunto existente, adicionando novas subclasses; oferece a possibilidade da criação de sistemas multiagentes, incluindo agentes reativos bem como deliberativos e utiliza o paradigma de programação orientada a agentes AOP.

Como característica negativa a linguagem de comunicação (KQML), da versão utilizada, mostrou-se insuficiente em alguns casos, pois várias performativas<sup>13</sup> ainda não tinham sido implementadas nesta versão. Por exemplo a performativa *<ontology>*, não tem efeito em uma mensagem KQML, deste modo, não pode-se trabalhar com o conceito de ontologia [GUA97]. O que influenciou diretamente o AM, que não pode ter todas as suas capacidades implementadas, e.g. a atualização das BD, são efetuadas pelo AMT e não pelo AM.

Neste capítulo mostrou-se a filosofia de operação do SISMAT, assim como, apresentaram-se exemplos de situações que envolvem a matrícula de um acadêmico, e o comportamento da sociedade de agentes do SISMAT diante destas situações.

Observou-se que a interface que o sistema apresenta é muito simples e não exige que o AU possua um treinamento específico para operação do sistema. Ou seja, basta o AU ser qualquer pessoa (acadêmico, professor, coordenador de curso, secretário, chefe de departamento, etc.) envolvida com a instituição de ensino onde o SISMAT encontra-se instalado. Pode-se dizer que a interface do SISMAT, está em conformidade com as idéias de Maes (1994): “um grande número de usuários destreinados farão uso dos computadores e estações de trabalho, num futuro muito próximo, tornando imperiosa a necessidade de mudança do paradigma de interação usuário-computador, o gerenciamento direto, o qual exige que o usuário inicialize e monitore a totalidade dos eventos” [MAES94].

Sob este contexto, no SISMAT o usuário apenas inicializa o processo, ficando a cargo dos agentes o processamento dos conhecimentos necessários para a montagem da matrícula de um acadêmico.

---

<sup>13</sup> Performativa é a palavra chave de uma mensagem KQML.

Diante destas abordagens é possível afirmar que, as habilidades que os agentes podem possuir e as características que podem ser atribuídas a eles, em analogia com atitudes dos humanos, tornam os computadores ferramentas cada vez mais poderosas e maleáveis, podendo ampliar a gama de problemas que podem ser resolvidos.

Os agentes do protótipo do SISMAT desenvolvido em LALO, não foram implementados na plenitude de suas funções, pelo fato de que todos os testes realizados foram efetuados em ambiente simulado, sendo assim as bases de dados utilizadas constituíram-se de dados de acadêmicos, cursos, históricos escolares, matrículas hipotéticas.

Contudo, a modelagem e a arquitetura desenvolvida para os agentes do SISMAT, torna possível em trabalhos futuros, a implementação do sistema em um ambiente real.

Tomando por base a concepção, projeto e modelagem do SISMAT chegou-se as seguintes conclusões.

## 6 Conclusões

Este trabalho apresenta uma aplicação prática das idéias de SMA e AOP em um ambiente complexo como o de matrículas em uma universidade. Pois neste ambiente, o volume de conhecimento é muito grande e as pessoas envolvidas em um processo de matrícula, não possuem individualmente todo o conhecimento para realizá-las. Sendo necessário um trabalho cooperativo e a busca de informações em uma base de dados.

O protótipo do sistema foi desenvolvido com a ferramenta de desenvolvimento de agentes LALO que possibilita a programação orientada a agentes. Todos os agentes que integram o sistema possuem características particulares e comunicam-se entre si através da linguagem de comunicação KQML.

Para obter a sociedade multiagente do SISMAT, duas principais tarefas tiveram que ser realizadas. A primeira foi a modelagem da sociedade como um todo onde foram definidos quantos e que tipo de agentes (reativos ou cognitivos) seriam necessários. Na etapa seguinte foi realizada a modelagem da arquitetura de cada agente, modelando-se os diferentes estados mentais e as características particulares, desta forma o comportamento individual dos agentes pode ser definido.

Neste contexto os agentes do SISMAT apresentam as seguintes propriedades: Autonomia, Comunicabilidade, Cooperação, Iniciativa, Reatividade, Sociabilidade e Reusabilidade. Sendo que, estas características somente puderam ser atingidas com a abordagem SMA.

A modelagem através de estados mentais, representa um aspecto importante no comportamento dos agentes do SISMAT, pois possibilita aos mesmos em determinadas condições, tomarem suas próprias decisões baseadas em seu estado mental, independentemente da interferência do usuário, o que atribui a estes um certo grau de autonomia na execução de suas tarefas.

Ainda no que tange os estados mentais e especificamente a atribuição de crenças aos agentes, pode ser destacado outro ponto positivo na implementação de agentes. Pois as crenças, permitem representar conhecimentos que ficam explícitos aos agentes e não ficam apenas implícitos nos códigos de programas que os implementa.

Além dos estados mentais, o fator tempo que é associado aos agentes, também possui uma importância significativa, pois permite criar a noção de lembrança, o que aumenta ainda mais a capacidade de decisão dos agentes, tendo a possibilidade de raciocinarem sobre fatos do presente, passado e futuro.

Outra característica de real importância que também deve ser destacada no desenvolvimento de agentes com AOP, é a comunicação e especificamente o uso do KQML. Pois possibilita aos agentes trocarem mensagens entre si, sendo que, o conteúdo destas reflete o estado mental do agente naquele momento. Permitindo ao agente uma maior flexibilização em seu raciocínio e tomada de decisões.

Como trabalho futuro propõem-se, uma ampliação do SISMAT, onde este além de efetuar matrículas, também possa fazer alocação de disciplinas, horários e ensalamento de professores.

Isto é algo ainda não conseguido satisfatoriamente através do uso do computador, devido a sua complexidade e exigir simultaneamente a interação de diversos elementos. Desta forma, utilizando os paradigmas AOP e SMA, poder-se-á verificar e avaliar qual o nível de solução que se obterá.

## Referências Bibliográficas

- [AGENB99] Web: <http://www.agentbuilder.com>, 1999.
- [BAR97] BARRETO, J.; *Inteligência Artificial no Limiar do Século XXI*, ISBN 85-900382-2-X Editor Jorge Muniz Barreto, PPP Edições, Florianópolis, 1997.
- [BEL95] BELGRAVE, M.; *The Unified Agent Architecture: A White Paper*, Web: [http://www.ee.mcgill.ca/~belmarc/uaa\\_paper.html](http://www.ee.mcgill.ca/~belmarc/uaa_paper.html), 1995.
- [BOND88] BOND, A. H. e GASSER, L.; *Reading in Distributed Artificial Intelligence*, Morgan Kaufmann, 1988.
- [BRA89] BRATMAN, M. *What is Intention*. In: COHEN, P; MORGAN, J; POLLACK, M. (Eds.). *Intentions in Communication*. [S.l.]: MIT Press, 1989. In: GIRAFFA, L. M. M.; *Uma arquitetura de tutor utilizando estados mentais*, Universidade Federal do Rio Grande do Sul, 1999. (Tese de doutorado).
- [BRA91] BRAZDIL, P.; GAMS, M.; SIAN, S.; TORGO, L. e VAN DE VELDE, W.; *Learning in distributed systems and multi-agent environments*. Machine Learning – EWSL-91, 1991.
- [BRZ97] BRAZIER, F. e DUNIN-KEPLICZ, B.; *Belief, intentions and desire*. International Journal of Cooperative Information Systems, 1997.
- [COR94] CORRÊA, M.; *A Arquitetura de Diálogos entre Agentes Cognitivos Distribuídos*. Rio de Janeiro: UFRJ, 1994. In: GIRAFFA, L. M. M.; *Uma arquitetura de tutor utilizando estados mentais*, Universidade Federal do Rio Grande do Sul, 1999. (Tese de doutorado).
- [COS93] COSTA, A. C. R.; *Inteligência de Máquina: Esboço de uma Abordagem Construtivista*. Phd thesis, Universidade Federal do Rio Grande do Sul, 1993.



- [COEN96] COEN, M.; “Sodabot Agent”, Web: <http://www.ai.mit.edu/people/sodabot>, 1996.
  
- [DEM90] DEMAZEAU, Y. e MULLER, J.; *Decentralized Artificial Intelligence*, First European Workshop on Modeling Autonomous Agents in a Multi-Agent World. Proceedings, Cambridge, North-Holland, 1990.
  
- [DEM92] DEMAZEAU, Y. e SICHMAN, J. e BOISSIER, O.; *When can knowledge-based systems be called agents?*, IX Simpósio Brasileiro de Inteligência Artificial, Rio de Janeiro (RJ), 1992.
  
- [ERC91] ERCEAU, J. e FERBER, J.; *L’Intelligence Artificielle Distribuée. La Recherche*, 22(233):750-758, 1991.
  
- [FER91] FERBER, J. e GASSER, L.; *Intelligence artificielle distribuée*. International Workshop on Expert Systems & Their Applications, 1991.
  
- [FRA96] FRANKLIN, S. e GRAESSER, A.; “*Is it an Agent. or just a Program?: A Taxonomy for Autonomous Agents*”, University of Memphis, Web: <http://www.msci.memphis.edu/~franklin/AgentProg.html>, 1996.
  
- [GIR99] GIRAFFA, L. M. M.; *Uma arquitetura de tutor utilizando estados mentais*, Universidade Federal do Rio Grande do Sul, 1999. (Tese de doutorado).
  
- [GIR97] GIRAFFA, L. M. M.; *Seleção e adoção de Estratégias de Ensino em Sistemas Tutores Inteligentes*. CPGCC/UFRGS, 1997. (EQ-10).
  
- [GUA97] GUARINO, N.; *Understanding, building and, using ontologies*. International Journal of Human-Computer Studies, 1997.

- [GRU93] GRUBER, T.R.; *A Translation Approach to Portable Ontology Specifications*. Technical Report KSL 92-71, Stanford University, Knowledge Systems Laboratory, 1993.
  
- [HAL92] HALPERN, J.Y.; MOSES, Y.; *A guide to completeness and complexity for modal logics of knowledge and belief*. Artificial Intelligence, Amsterdam, v.54, p.319-379, 1992. In: GIRAFFA, L. M. M.; *Uma arquitetura de tutor utilizando estados mentais*, Universidade Federal do Rio Grande do Sul, 1999. (Tese de doutorado).
  
- [HAY99] HAYES, C. C.; *Agents in a Nutshell – A Very Brief Introduction*. IEEE Transactions on Knowledge and Data Engineering – IEEE Computer Society, 1999.
  
- [HEI95] HEILMANN, K.; KIHANYA, D.; LIGHT, A. e MUSEMBYA, P.; *Intelligent Agents: A Technology and Business Application Analysis*. Web: <http://haas.berkeley.edu/~heilmann/agents/index.html>, 1995.
  
- [HUB98] HÜBNER, J. F. e COELHO, A. R.; *Linguagens e protocolos de Comunicação entre Agentes*. Relatório Final Projeto PIBIC/CNPq. FURB - Universidade Regional de Blumenau, 1998.
  
- [LAB97] LABROU, Y.; *Semantics for an Agent Communication Language*. Phd thesis, University of Maryland Graduate School, Baltimore, Maryland 21228-5398, 1997.
  
- [LALO97] LALO (*Langage of Agents Logiciel Objet*), CRIM - Centre de Recherche Informatique de Montreal. Web: <http://www.crim.ca/sbc/english/lalo>, 1997.
  
- [LESS99] LESSER, V. R.; *Cooperative Multiagent Systems: A Personal View of the State of the Art*. IEEE Transactions on Knowledge and Data Engineering – IEEE Computer Society, 1999.

- [MAES91] MAES, P.; *The agent network architecture (ana)*. SIGART Bulletin, 1991.
- [MAES94] MAES, P.; *Agents that Reduce Work and Information Overload*, Web: <http://pattie.www.media.mit.edu/people/pattie/CACM-94/CACM-4.P1.html>, 1994.
- [OLI96] OLIVEIRA, F.; *Inteligência Artificial Distribuída*, In: IV Escola Regional de Informática, SBC, SC, 1996.
- [PIER96] PIERITZ, H. I.; *Linguagens - A Base para a Construção de Agentes Inteligentes*, Web: <http://www.inf.ufsc.br/iad/users/h/homero/linguagens.htm>, 1996.
- [POZO96] POZO, A. T. R.; *Um Sistema de Ensino Inteligente via Sociedade Multiagentes aplicado ao Diagnostico de Epilepsia*, Universidade Federal de Santa Catarina, 1996. (Tese de doutorado).
- [RIV96] RIVER, C.; “*Agent Technology*”. Web: <http://www.opensesame.com/webpages/sesame/whitepaper.html>. 1996.
- [RUS95] RUSSEL, S. e NORVIG, P.; *Artificial Intelligence: a modern approach*. Prentice-Hall, Englewood Cliffs, NJ, 1995.
- [SCHB94] SCHREIBER, G.; WIELINGA, B.; HOOG, R. et al.; “*CommonKADS: A Comprehensive Methodology for KBS Development*”, IEEE EXPERT, pag. 28-36, 1994.
- [SHO93] SHOHAM, Y.; *Agent-oriented programming*. Artificial Intelligence, Elsevier, Amsterdam, v.60, n.1,(51-92),1993.
- [SOU96] SOUZA, E. M. S.; *Uma Estrutura de Agentes para Acessoria na Internet*, Dissertação de Mestrado, UFSC, PEPS, Brasil. 1997.

- [UEPG99] Catálogo Geral 99 – Universidade Estadual de Ponta Grossa – Pró-Reitoria de Graduação – Divisão de Ensino, 1999.
- [VIR95] VIRDHAGRISWARAN, S.: “*Mobile Unstructured Business Object Technology*”, Web: <http://www.crystaliz.com/logicware/mubot.html>, 1995.
- [WEISS99] WEISS, G.; *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. Massachusetts Institute of Technology, ISBN 0-262-23203-0, 1999.
- [WOOD94] WOOD, A.; *Agent-based interaction*. Phd. Progress Report PR-94-4, 1994.
- [WOOL95] WOOLDRIDGE, M. J. e JENNINGS, N. R.; *Intelligent Agents: Theory and Practice*. Knowledge Engineering Review v.10 No 2, United Kingdom, 1995.
- [WOOL92] WOOLDRIDGE, M.; *On the logical Modelling of Computational Multiagent Systems*. Manchester: UMIST, Department of Computation, 1992. In: GIRAFFA, L. M. M.; *Uma arquitetura de tutor utilizando estados mentais*, Universidade Federal do Rio Grande do Sul, 1999. (Tese de doutorado).
- [WOOL96] WOOLDRIDGE, M. J. e JENNINGS, N. R.; *Agent Theories, Architectures, and Languages: A Survey*, United Kingdom, 1996.